NR ind. 366196

MOJE Atari

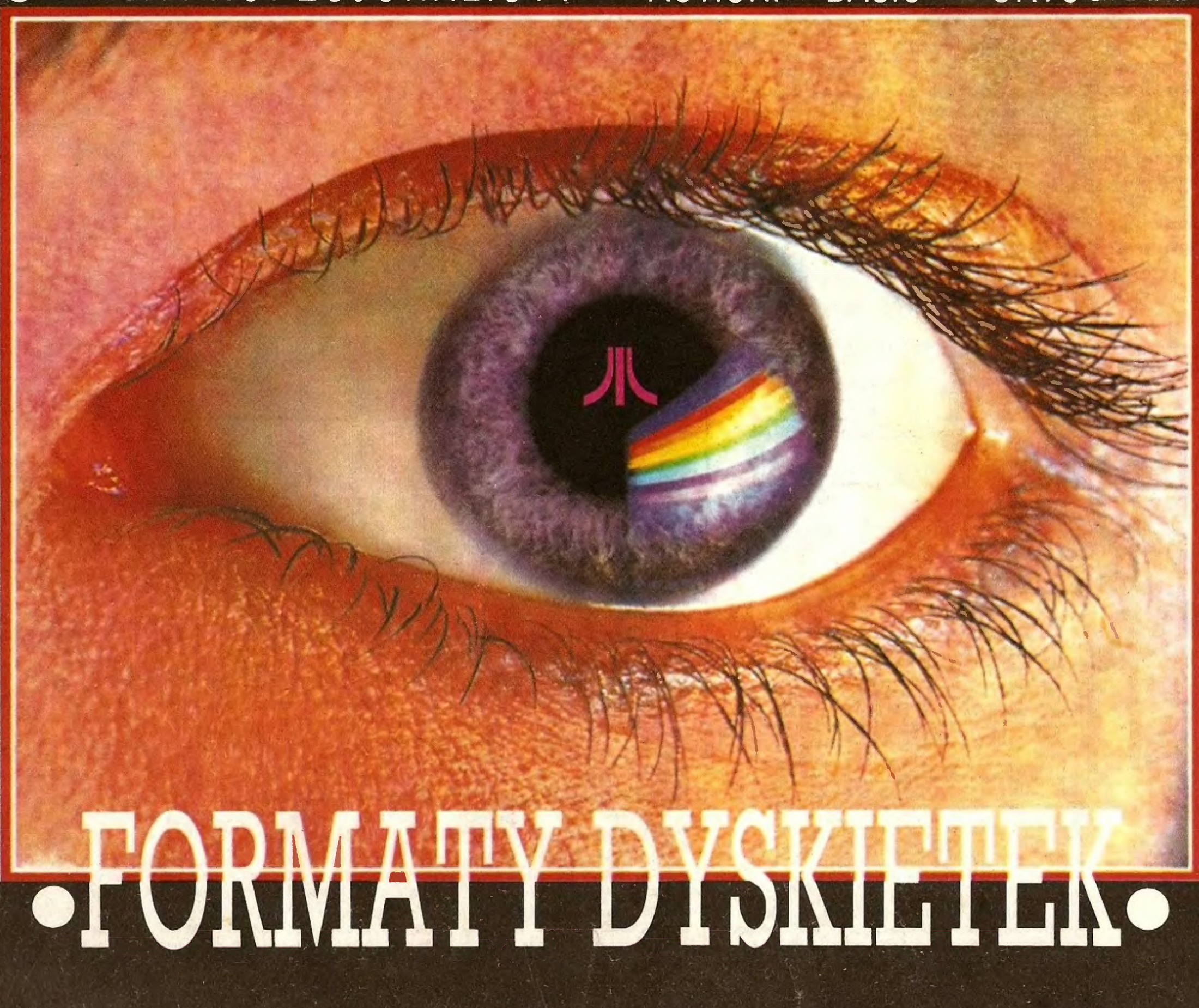
CENA 8000 ZŁ

MAGAZYN UŻYTKOWNIKÓW KOMPUTERÓW ATARI • LIPIEC/SIERPIEŃ'91 (6'91)

WIELKI KONKURS!

6

ST DLA PROFESJONALISTY • ACTION! + BASIC • ORTOGRAFIA • WOJNY RDZENIOWE • JAK KODOWAĆ?



ZAPIS I ODCZYT RYSUNKÓW KOALI • TOP 12

•FORMATY DYSKIETEK•

CO? w numerze:

Drogi Czytelniku!

Wakacje już za nami i każdy z pewnością podsumowuje bilans zysków i strat. Do zysków można zapewne zaliczyć regenerację sił do pracy lub nauki oraz różnorodne przeżycia wakacyjne, a czasem także nowe nabytki: programy, komputery i akcesoria. Straty są głównie natury finansowej, gdyż niemal każdy wakacyjny wyjazd znacznie zubaża zawartość kieszeni (oczywiście poza uprawiającymi „turystykę handlową"). W tym kontekście dodatkową nieprzyjemnością staje się fakt zwiększenia ceny naszego pisma. Nie będę jednak tłumaczył, że to wzrost cen papieru , druku itd, itp. Myślę, że do tego wszyscy się już przyzwyczaili. Zamiast takich wyjaśnień należą się przede wszystkim bardzo gorące przeprosiny, iż zmiana ta została dokonana i na to — na dodatek — bez żadnego uprzedzenia. Mogę się tylko usprawiedliwiać tym, że sam dowiedziałem się o podniesieniu ceny dopiero podczas druku numeru 5.

Równie serdecznie przepraszamy za opóźnienie w ukazywaniu się naszego pisma. Zgodnie z zapowiedzią powinno się ono bowiem ukazać w pierwszych dniach czerwca. Niestety, najpierw ważniejszy okazał się druk broszur, czy też ulotek (dokładnie nie sprawdzałem) dla nowej przewodniej siły narodu, a następnie zepsuła się w drukarni maszyna. Jak zapewne zauważyłeś spowodowało to załamanie się harmonogramu wydawania wszystkich pism naszej spółdzielni. Możemy tylko obiecać, że postaramy się, aby więcej się to nie zdarzyło. Na marginesie dodam, iż prowadzimy rozmowy z innymi drukarniami w sprawie druku naszych pism. Będzie się to oczywiście wiązało z przejściem na skład komputerowy, zmianą formatu i przede wszystkim poprawą jakości. Rozmowy te są bardzo trudne, gdyż ceny oferowane przez inne drukarnie są znacznie wyższe, a nie chcemy nadmiernie obciążać Twojej kieszeni. Może jednak już wkrótce zakończą się powodzeniem, czego życzy Ci (i sobie)

Wojciech ZIENTARA

PROGRAMOWANIE

Action! + Basic 15
Program w Action! można połączyć z napisanym w każdym innym języku. Jak to zrobić? Korzystając z przykładów współpracy Action! z Turbo Basicem.

Duszki w natarciu 12
O grafice graczy i pocisków napisano już bardzo dużo i niemal każdy użytkownik wie, jak ją projektować. Ale po co robić to samemu, jeśli można zatrudnić komputer.

Tylko dla zaawansowanych (3) 13
W poradach dla programistów tym razem o tajnikach pracy magnetofonu i jego współpracy z komputerem.

GRY

Wojny rdzeniowe 6
Bardzo znana i popularna gra wojenna nareszcie w wersji na „małe" Atari. Granie w nią wymaga jednak ogromnej cierpliwości i trochę umiejętności programowania.

Nowe i stare 23
Opisy gier „Taipei XL", „Decision in Desert", „Saracen", „Fortress", „Crystals of Arborea" i „Lost Patrol".

TOP 12 22

PROGRAMY UŻYTKOWE

Katalog dyskietki w Basicu 8
Zainstalowanie w Atari Basic nowej instrukcji DIR pozwala na odczyt katalogu dyskietki bez przechodzenia do DOS-u. Szczególnie przyda się to właścicielom 800XL i 65XE.

Zapis i odczyt rysunków Koala 20
Program Koala zapisuje swoje rysunki w specjalnym, skondensowanym formacie. Przy pomocy pokazanych tu procedur można ten format zastosować we własnych programach.

SPRZĘT

Ochrona przed zapisem w stacjach LDW i CA 8
Zapis dyskietki niezależny od wycięcia w niej tym razem dla stacji LDW Super 2000 i California Access 2001.

Systemy instalowane w stacjach dysków 14
Charakterystyka oryginalnej stacji Atari 1050 i dwóch pierwszych, instalowanych w niej systemów: Turbo 1050 i Duplicator 1050.

Atari jako sterownik 17
Komputer Atari może być wykorzystany jako programowalny sterownik dowolnych urządzeń elektrycznych. Nie trzeba w tym celu w nim nic zmieniać, wystarczy wykonać prosty interfejs.

SZKÓŁKA

Kodowanie (3) 10
Przystępujemy do zamiany na program zasadniczej części algorytmu naszej gry. Jeszcze trochę i nareszcie zacznie ona działać.

Formaty dyskietek 18
Znając budowę dyskietki możemy zająć się jej działaniem. Pierwszą operacją jest zawsze formatowanie — tu dowiesz się po co i w jaki sposób.

INSTRUKCJE

Crusade in Europe 19
W „Vademecum Atari" krótki poradnik korzystania z doskonałej gry strategicznej firmy Microprose — „Krucjata w Europie", czyli lądowanie w Normandii i wyzwolenie Francji.

DZIAŁ ST

ST z monitorem Amstrada 27
Zamiast kupować drogi monitor kolorowy do naszego ST lepiej wykorzystać stary monitor od Amstrada. Trzeba tylko zrobić odpowiedni kabelek.

Proste programy 30
Nasi czytelnicy nadesłali pierwsze programy na ST. Jak zwykle na początku tematem ich są zagadnienia matematyczne.

Emulator Macintosha 31
Atari ST można ulepszać prawie bez końca. Można też przerobić go niemal na każdy inny komputer. Aby Atari ST stał się Macintoshem, potrzebna jest tylko niewielka przystawka.

ST dla profesjonalisty 32
Przemysłowe zastosowanie ST? Tak, to możliwe, jeśli jest to 190 ST 020 firmy IBP.

INNE

Krótkie, krótsze i najkrótsze 11
Rozpoczynamy nowy cykl, w którym będziemy publikować krótkie programiki nadesłane przez naszych Czytelników.

Atari na CeBICie 28
CeBIT to największe na świecie targi komputerowe. Co nowego pokazała na nich w tym roku firma Atari? Przeczytaj!

Ortografia 16
Recenzja doskonałego programu edukacyjnego do nauki poprawnej pisowni. Tylko dla Atarii XL i XE.

Atari Logo 17
Logo dla Atari XL/XE jest najlepszą implementacją tego języka na ośmiobitowym komputerze. Najlepszym podręcznikiem do niego jest natomiast nowa książka Ryszarda Tadeusiewicza.

Wielki konkurs 16

Wieści 3

Listy 4

Edytor Basica 5

Wpisywanie znaków specjalnych 5

UWAGA: Wszystkie programy opublikowane w naszych wydawnictwach o Atari („Tylko o Atari" 1 i 2 oraz „Moje Atari" 1–6) można nabyć na dyskietkach dostępnych w cenie 15000 zł w redakcji: Warszawa, ul. Wasilkowskiego 7, wtorek-czwartek, godz. 10–14. Ponadto dyskietki te są sprzedawane przez sklepy „Bajtka" w Krakowie i Bytomiu oraz niektóre inne sklepy komputerowe.

MOJE Atari

Redaktor naczelny:
Wojciech Zientara
Sekretarz redakcji:
Tomasz Wiśniewski
Opracowanie graficzne:
Wanda Roszkowska
Redaktor techniczny:
Maria Radzimińska
Zdjęcia:
Leopold Dzikowski

Redakcja:
02-776 Warszawa, ul. Wasilkowskiego 7
tel. 643-18-40

Wydawca:
Spółdzielnia „Bajtek"
00-687 Warszawa, ul. Wspólna 61
tel. 21-12-05

Skład i druk:
Przedsiębiorstwo Wydawniczo-Poligraficzne "GRYF"
Ciechanów, ul. Sienkiewicza 51
Fotoskład:
Wiesława Mroczkowska
Montaż offsetowy:
Grażyna Ostaszewska
Korekta:
Maria Krajewska
Teresa Rutkowska
Nr zlecenia: 86430
Nakład: 105 tys. egz.
ISSN 0867-390X

Redakcja zastrzega sobie prawo skracania i adjustacji nadesłanych artykułów. Materiałów nie zamówionych nie zwracamy. Za treść ogłoszeń redakcja nie odpowiada.

WIZYTÓWKA DLA KAŻDEGO

Na dworcach kolejowych i lotniskach w RFN ustawione kolorowe automaty, które sprzedają bilety, informują o przylotach i odlotach samolotów, przyjazdach pociągów. Są wśród nich także urządzenia świadczące podróżnym inne usługi.

Pasażerowie przechodzący przez halę dworca kolejowego w Düsseldorfie lub perony szybkiej kolei miejskiej we Frankfurcie nad Menem mogą w ciągu kilku chwil zaopatrzyć się w wizytówki. Komputerowy automat zapewnia błyskawiczną obsługę.

Przednia, pionowa ścianka urządzenia jest ozdobiona napisem „FIXCARD — wizytówka dla każdego". Obok widnieje kilkanaście wzorów biletów wizytowych i paru druków okolicznościowych. Tuż poniżej wystaje lekko pochyły pulpit z czarnobiałym monitorem oraz klawiaturą wykonaną z plastykowej folii, takiej jak w ZX81 i wielu kieszonkowych kalkulatorach. Napis na ekranie zachęca do wrzucenia 7 DM. Automat połyka monety i możemy wybrać wzór wizytówki, która najbardziej nam się podoba. Teraz należy wystukać na klawiaturze swoje dane personalne: imię i nazwisko, adres, numer telefonu, stanowisko. Po chwili z urządzenia wydobywają się przytłumione dźwięki, zaczyna pracować ukryta wewnątrz drukarka. Jeszcze troszkę cierpliwości i stajemy się posiadaczami czterdziestu całkiem przyzwoicie wyglądających wizytówek.

Klawiatura na pulpicie jest zaprojektowana z myślą o osobach, które mają bardzo mgliste pojęcie o komputerach. Zamiast nadruku „Backspace" widnieje na klawiszu napis „Korekta", „Space" nazywa się „Odstęp", a „Return" oznaczono „Dalej", wszystko oczywiście po niemiecku.

Program obsługujący automat napisano w GFA-Assemblerze. Dyskietka jest umieszczona na stałe w napędzie. Po zaniku zasilania nie trzeba wołać fachowców z serwisu, program uruchamia się samoczynnie. Oprócz 1040 ST ze stacją dysków zamontowano wewnątrz automatu monitor SM 124 i drukarkę.

Producent urządzenia nie chce ujawniać bliższych danych technicznych drukarki. Można na ten temat dowiedzieć się tylko, że jest to drukarka termiczna z rolką papieru o odpowiedniej grubości i szerokości. Każda wizytówka jest zaraz po wydrukowaniu równiutko odcinana z rolki.

Pomyślano również o zabezpieczeniu przed chuliganami. Atari 1040 ST stanowi łakomy kąsek dla różnych opryszków, ale obudowa i przezroczysta szyba osłaniająca monitor na pulpicie są wykonane z solidnego materiału. Pokonanie tej przeszkody jest możliwe tylko przy pomocy potężnego łomu.

(J)

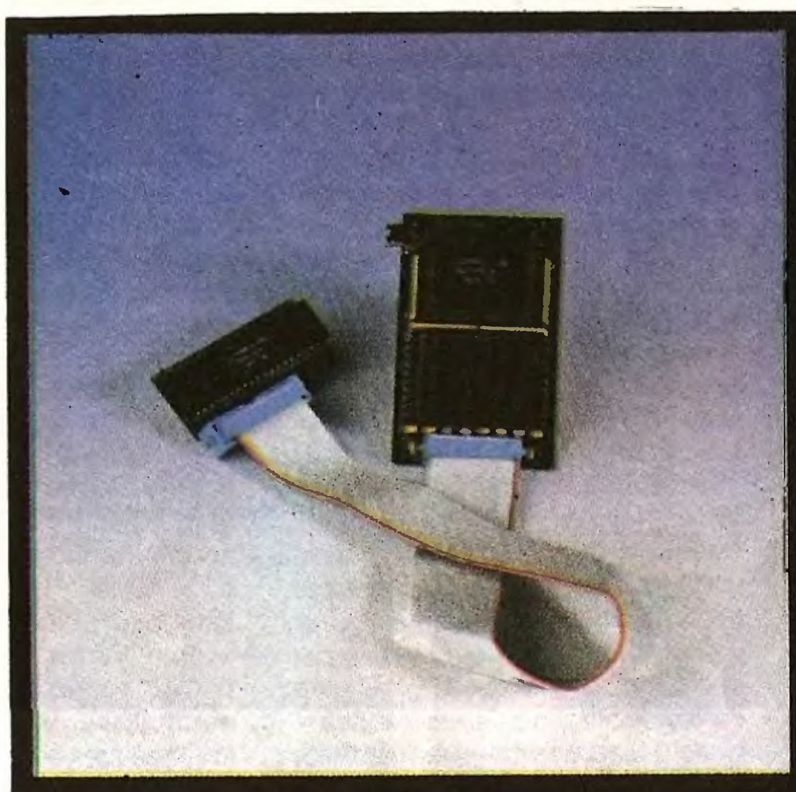
STACJA DYSKÓW 1,44 MB DO ST

Im wyższe wymagania są stawiane przed komputerem, tym bogatsza musi być jego konfiguracja. W pewnych przypadkach ST ze standardową stacją dysków elastycznych 720 KB może nie spełniać wszystkich oczekiwań użytkownika. Napędy o tej pojemności od dawna nie są najnowocześniejszym rozwiązaniem technicznym. Ale jest na to rada.

Firma Compuware oferuje pakiet UK1, który umożliwia przebudowanie komputera. Jest to zmodyfikowana 3,5" stacja dysków do Atari ST/E, Mega ST/E i Atari TT oraz Tower \Pojemnośc napędu w tej wersji wynosi 1,44 MB. Posiada on możliwośc automatycznego rozpoznawania rodzaju stosowanej dyskietki oraz potrafi czytać i zapisywać w formacie MS-DOS.

Pakiet UK1 jest sprzedawany wraz z podręcznikiem, instrukcją montażu i oprogramowaniem służącym do formatowania i kopiowania dyskietek. Cena w RFN wynosi około 298 DM.

Na podst. ST Magazin 6/91

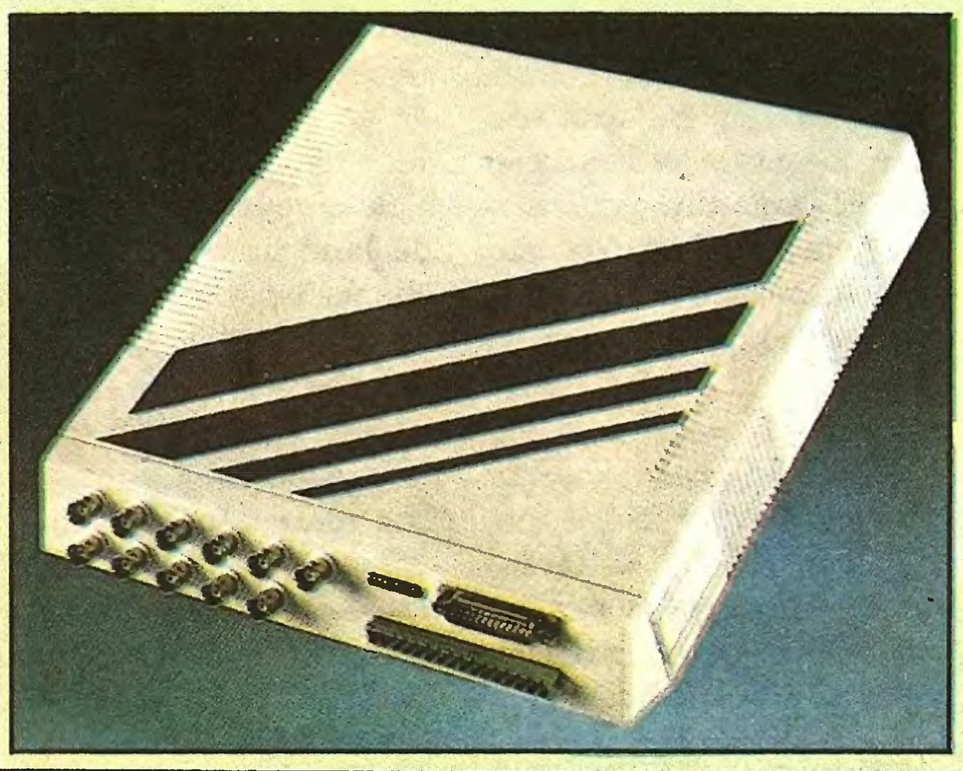


ROZSZERZENIE PAMIĘCI

Rozszerzenia pamięci komputera oferuje bardzo wiele firm. Na zdjęciu prezentujemy jedno z takich urządzeń, którego dystrybutorem w RFN jest Heim Verlag. Płytka ma niewielkie rozmiary, jest wykonana bardzo estetycznie i łatwa w montażu. Jej zainstalowanie nie koliduje z innymi dodatkowymi kartami ST, np. z emulatorem AT-Speed, kartą Hypercache lub MegaScreen. Użytkownik może ją samodzielnie zamontowac w komputerze, w czym pomaga mu dobrze opracowana instrukcja. Dołączane do zestawu oprogramowanie pozwala sprawdzić całą pamięć komputera i wykryć uszkodzone układy.

Jeszcze przed paru laty domowy komputer z 64 KB pamięci był absolutną rewelacją. Obecnie w wielu zastosowaniach niezbędne jest kilka megabajtów. Nic dziwnego, że użytkownicy Atari 520 ST i 1040 ST poszukują rozwiązań pozwalających tanio i szybko rozbudować posiadany komputer.

Rozszerzenie jest oferowane w dwóch wersjach: 2,5 oraz 4 MB, które kosztują od 598 do 798 DM. Urządzenie to zyskało wiele pochlebnych opinii. Jego teksty ukazały się w ST-Computer 12/90 i miesięczniku Ct 1/91



ATARI JAKO MIERNIK

Dzięki urządzeniu E-Labor firmy Rothron każde Atari ST może być wykorzystywane jako urządzenie sterujące i pomiarowe. E-Labor jest wielofunkcyjnym interfejsem, który przyłącza się do komputera 520, 1040 ST lub Mega ST poprzez User-Port lub szynę systemową.

Producent przewiduje liczne możliwości zastosowania tego interfejsu w szkolnictwie lub w domu. Urządzenie dysponuje wieloma analogowymi i cyfrowymi wyjściami. E-Labor może zastąpić różne urządzenia pomiarowe i pozwala zaznajomić się ze wspomaganymi komputerowo procesami pomiarowymi, sterowania lub regulacji.

Przy pomocy interfejsu można mierzyć napięcie, natężenie prądu, częstotliwość oraz wykorzystywać komputer jako oscyloskop. E-Labor kosztuje w zależności od wersji 498–998 DM.

(Na podst. ST Magazin 6/91)

opracował: Janusz Jarmoch

ST TY

Na listy Czytelników odpowiadają Wojciech Zientara i Janusz Jarmoch

Czy drukarka Star LC-10 współpracuje z 1040ST bez interfejsu? Czy wszystkie programy graficzne na ST współpracują z LC-10?

Gdzie można kupić i ile kosztuje skaner do ST?

Co to są programy "Desktop Publishing" i gdzie można nabyć program "Calamus"?

Czy do LC-10 potrzebny jest specjalny papier i taśma, a jeżeli tak, to gdzie można je kupić?

Michał Murzynowski
Gdańsk

Atari ST posiada wbudowane interfejsy Centronics i RS232, a więc współpracuje ze wszystkimi drukarkami wyposażonymi w takie złącza. Nie można do niego przyłączyć jedynie drukarek specjalnie przystosowanych do niektórych komputerów, np. Spectrum, Commodore, Atari XL itp. Większość programów graficznych umożliwia stosowanie różnych drukarek. Są one wybierane z menu lub poprzez określenie procedury obsługi (*drivera*) drukarki. Specjalne skanery do ST można nabyć jedynie na Zachodzie, a także w firmach, które sprowadzają je do Polski. Można jednak zastosować inne skanery (do IBM) pod warunkiem, że nie mają one karty instalowanej wewnątrz komputera.

Programy typu Desktop Publishing służą do składu i druku komputerowego ulotek, plakatów, czasopism, broszur, książek itd. W Polsce licencjonowane kopie Calamusa rozprowadza Studio Stefana Szczypki.

Do drukarki LC-10 można stosować dowolny papier dobrej jakości. Taśma jest natomiast w specjalnej kasecie. Można ją nabyć w większości sklepów ze sprzętem komputerowym. Ponadto zużytą taśmę można zregenerować poprzez nasączenie specjalnym tuszem. Usługę tą wykonują liczne specjalistyczne zakłady.

***

Niedawno dostałem od kolegi „Bajtki" z minionych trzech lat. W jednym z numerów „Tylko o Atari" został umieszczony artykuł „Mam Atari". Zostało w nim napisane „Atari ma lepszą grafikę, Commodore — dźwięk". Wątpię, czy jest to prawdą. Dziwię się Panu i wielu innym, którzy myślą, że Atari ma lepszą grafikę, niż Commodore. Za prosty przykład niech posłuży znana gra „World Karate Championships". Na Atari po pokonaniu kilku zawodników nie miałem podanego koloru pasa i nie zmieniało się tło. Na Commodore ta gra jest urozmaicona: zmienia się kolor pasa, zmienia się tło. Myślę, że tym przykładem przekonałem Pana, że Commodore ma lepszą grafikę niż Atari.

Grzegorz Płaczek
Tarnowskie Góry

Najpierw zapytam, gdzie została kupiona gra „World Karate"? Z pewnością od złodzieja, gdyż oryginalna dyskowa wersja tego programu posiada wszystkie wymienione elementy. Pirackie kopie są ich pozbawione, aby uprościć przeniesienie programu na kasetę, a interes użytkownika jest zupełnie pomijany — i tak przecież kupi on wszystko. Przekazane w liście uwagi dotyczą przecież tylko liczby plansz, a nie ich wyglądu i nie mogą być argumentem rozstrzygającym o możliwościach komputera — mówią one tylko o pracy programistów. Grafika to jednak nie tylko to, co można zobaczyć w kilku programach, lecz to, co można rzeczywiście uzyskać na ekranie. Programiści gier piszą je na kilka komputerów równocześnie, więc robią to w najprostszy możliwy sposób. Możliwości graficzne Atari są rzeczywiście większe, ale trudniej je w pełni wykorzystać. W ten sposób lenistwo programistów (i wygodnictwo złodziei kradnących i „poprawiających" te programy) psuje opinię o komputerze.

***

Po nabyciu „Mojego Atari" zacząłem wklepywać Wasze programy użytkowe — robota dość żmudna i dająca wątpliwe efekty, jeśli na koniec okazuje się, że w listingu jest błąd. Mogę zrozumieć, że to wina drukarni. To jeszcze mogę przełknąć, ale, gdy w numerze 4 przeczytałem Waszą erratę, poniosło mnie. Piszecie, że raczej nie będziecie drukować poprawionych programów. Co w takim razie zostanie w numerze, jeśli programy są do kitu?

W sklepach w Łodzi powiedziano mi, że stacje dysków do Atari są niedostępne ani w Polsce, ani w RFN. Czy to prawda? Czy Atari staje się sprzętem martwym?

Błażej Powroźnik
Radomsko

Na podstawie „Erraty" zamieszczonej w „Moim Atari 4" można poprawić zarówno „Okna", jak i „MultiDOS". Należy wszakże przy ich przepisywaniu korzystać z „Edytora Basica" — po to właśnie drukujemy go w każdym numerze. Ponowne drukowanie całych programów zajęłoby tylko miejsce przeznaczone na inne artykuły.

Stacje dysków do Atari można kupić, np. w „Pewexie", ponadto bardzo dobre stacje są produkowane i sprzedawane przez firmę TOMS. Są one również dostępne w RFN, lecz trzeba ich szukać, gdyż rzeczywiście na Zachodzie o sprzęt ośmiobitowy jest już bardzo trudno.

***

Trafiłem na problem przerastający moje możliwości. Niedawno otrzymałem kłopotliwy prezent w postaci dwóch drukarek: Facit (rozetkowa z wałkiem 70 cm) i Epson MX82/T (9-igłowa, format A4 z traktorem). Żadna z nich nie posiada instrukcji ani przewodu sieciowego. Metodą prób udało mi się zmusić Epsona do druku alfabetu i na tym koniec. Podejrzewam, że drukarka jest ostatnią rzeczą, jaką kupują właściciele ośmiobitowców. Może dlatego nikt z moich znajomych nie umiał mi pomóc. Jak połączyć Atari z Epsonem? Jakie sygnały są konieczne do obsługi? Czy SpeedScript potrafi obsłużyć taki zestaw? Jak ustawić DIP-y dla prawidłowej konfiguracji? Drukarka jest wyposażona w dwa gniazda. Jedno z nich wygląda jak pomniejszony negatyw szyny krawędziowej Atari (Cenronics?). Drugie natomiast to 25 otworów w dwóch rzędach (RS232?). Czy drukarka posiada wbudowany jakiś interfejs?

Jednocześnie chciałbym pomóc wszystkim tym, którzy mają kłopoty z podłączeniem ukochanego komputerka do odbiornika telewizyjnego wyposażonego jedynie w złącza EURO (SCART). Niektóre typy odbiorników mają nawet dwa takie złącza, co nie ma większego znaczenia, gdyż oba mogą pracować jako wejścia. Gniazdo EURO ma kształt wydłużonego prostokąta z wystającym jednym narożnikiem. Numeracja styków rozpoczyna się od narożnika leżącego dokładnie naprzeciw (po przekątnej) od narożnika wystającego. Styki są numerowane naprzemian tak, że w jednym rzędzie znajdują się numery nieparzyste, a w drugim (od strony wystającego narożnika) parzyste. Interesujące nas sygnały są rozmieszczone następująco:

- ***2 — wejście fonii, kanał prawy (10 kΩ, 0,5V)***
- ***6 — wejście fonii, kanał lewy (10 kΩ, 0,5V)***
- ***20 — wejście całkowitego sygnału wizji (1V, ±3 dB, 75 Ω)***
- ***21 — masa***

Nie podaję, jak podłączyć to gniazdo z konkretnym komputerem, ale wydaje mi się, że każdy jest w stanie wykonać takie złącze sam.

Marek Tymoszczuk
Biała Podlaska

Z opisu złączy drukarki Epson wynika, że istotnie posiada ona dwa interfejsy — Centronics i RS232 (rozpoznanie było prawidłowe). Niestety, Atari XL/XE nie posiada żadnego z tych interfejsów i trzeba je zakupić oddzielnie (proponuję Centronics). Wystarczy to do drukowania przy pomocy większości programów użytkowych, które są wyposażane w procedury współpracy z drukarkami firmy Epson. Dokładne określenie kodów sterujących niezbędnych do wywołania różnych funkcji drukarki będzie jednak wymagało eksperymentowania. Może w tym pomóc instrukcja od dowolnej innej drukarki, gdyż prawie wszystkie urządzenia tego typu są w dużym stopniu zgodne programowo z Epsonem (producenci podają nawet, że są *Epson compatible*). A może ktoś z czytelników posiada podobną drukarkę i mógłby udzielić pomocy (np. wypożyczając instrukcję lub przesyłając do redakcji kserokopię jej *Quick Reference*)?

***

W grudniowym numerze „Mojego ATARI" znalazła się krótka informacja o niemieckim czasopiśmie „ATARI Magazin". Po jej przeczytaniu postanowiłam napisać do redakcji tego pisma. List wysłałam w grudniu. 12.2.1991 r. otrzymałam średnich rozmiarów pakunek, w którym znalazłam odpowiedź od redakcji (przepraszam, byłej redakcji) „ATARI Magazin". Było tam napisane, iż pismo to zostało zlikwidowane w 1989 roku. W pakunku znajdował się ostatni numer tego czasopisma z grudnia tamtego roku.

Dorota Dziedzic
Płock

O likwidacji czasopisma „ATARI Magazin" oczywiście nie wiedzieliśmy. Nie opublikowalibyśmy wzmianki na jego temat, jeśli informacja ta dotarłaby do nas wcześniej. Magazyn ten kupowaliśmy tylko sporadycznie, głównie korzystając z uprzejmości znajomych wyjeżdżających za granicę. W sumie w redakcji zgromadziliśmy kilka numerów „Atari Magazin". Niestety zniknięcie pisma umknęło naszej uwadze.

Podobny los spotyka corocznie kilkadziesiąt tytułów ukazujących się w RFN. Takie są prawa rynku, które można też poznać, uważnie obserwując nasze kioski. „Atari Magazin" w dużej części poświęcony był 8-bitowemu Atari. W RFN zdecydowana większość użytkowników przestawiła się już na komputery 16-bitowe. Redakcje czasopism komputerowych muszą trzymać rękę na pulsie i dopasowywać się do wymagań czytelników. Coraz nowocześniejszy i tańszy sprzęt wyższej klasy sprawił, że pisma dla 8-bitowców giną.

(J.J.)

COMMODORE C64/128
ATARI 800XL,65/130XE
I INNE
Twój komputer zarobi na Ciebie i Twoją rodzinę
3–8mln zł miesięczn.
Informacje w Poradniku przesyłanym za zaliczeniem poczt.
27000zł przy odbiorze
Robert Norton
39-303 Mielec
skr.poczt.1

B146

TOMS wita WAS!

NOWOŚCI FIRMY TOMS

W niniejszym odcinku ogłoszeniowym polecamy:

Bardzo tanie stacje dysków 5,25" dla ST i AMIGI w wielu odmianach, z dodatkowymi funkcjami. Takich stacji nie proponuje nikt inny!

Samplery dla ST i AMIGI. Sampler ST posiada oryginalne polskie oprogramowanie!

Ponadto informujemy, że właśnie został opracowany interfejs RS232 przeznaczony do dobudowania do stacji TOMS 720 (test w MA 91/5). W związku z tym oferujemy i taką wersję naszej stacji!

Oczywiście polecamy także wszystkie wcześniejsze nasze opracowania, jak oryginalne rozszerzenia pamięci, usprawnienia stacji dysków itd, opisywane w Bajtkach i Moim ATARI.

Nasza firma gwarantuje solidność, niezawodność i wysoki poziom techniczny.

Nasz adres: TOMS, ul. Widok 14/1, 00-023 Warszawa
Tel. (0-22) 27-16-01, (0-2) 641-54-29.
Zapraszamy w godzinach 9–15.

EDYTOR BASIC-a

Przy przepisywaniu programów w Basicu nie sposób ustrzec się błędów.

Aby uniknąć żmudnego wyszukiwania popełnianych omyłek wszystkie programy w Basicu są publikowane wraz z kodami kontrolnymi. Zamieszczony obok program służy do kontroli tych kodów podczas przepisywania programu.

Wydruk „Edytora Basica" należy dokładnie przepisać i zapisać na kasecie lub dyskietce (najlepiej instrukcją LIST „C:" lub LIST „D:EDYTOR.LST"). Poprawność przepisania można sprawdzić samym „Edytorem" w opisany niżej sposób.

Przystępując do wpisywania dowolnego programu z naszego pisma trzeba najpierw wczytać i uruchomić „Edytor Basica". Następnie należy przepisywać kolejne wiersze programu. Po wpisaniu każdego wiersza i naciśnięciu klawisza RETURN wiersz ten pojawia się w dolnej części ekranu wraz z obliczonym kodem kontrolnym. Jeżeli wyświetlony kod jest taki sam, jak wydrukowany przed numerem wiersza, można przystąpić do wpisywania następnego wiersza. Jeśli kody są różne, to ponowne naciśnięcie RETURN wyświetla wpisany wiersz w górnej części ekranu i umożliwia dokonanie w nim niezbędnych poprawek.

Samo naciśnięcie RETURN wywołuje zawsze ostatnio wpisany wiersz. W celu wywołania innego, wcześniej napisanego wiersza trzeba podać jego numer poprzedzony gwiazdką (np. 1000) i nacisnąć RETURN. Wpisanie samej liczby powoduje wymazanie z pamięci komputera wiersza programu o takim numerze.

Po poprawnym przepisaniu całego programu trzeba przerwać pracę „Edytora Basica" przez naciśnięcie klawisza BREAK lub RESET. Następnie w celu usunięcia „Edytora" i zbędnych zmiennych (z tablic nazw i wartości zmiennych) zapisujemy program na kasecie instrukcją LIST „C:",0,31999 lub na dyskietce instrukcją LIST „D:nazwa",0,31999. Teraz kasujemy zawartość pamięci komputera instrukcją NEW i odczytujemy program przy pomocy ENTER „C:" (z kasety) lub ENTER „D:nazwa" z dyskietki. Po wykonaniu tych czynności w pamięci komputera znajduje się tylko gotowy program bez „Edytora Basica" i można go już ostatecznie zapisać na odpowiedni nośnik.

Procedura ta jest może nieco kłopotliwa, lecz zabezpiecza w stu procentach przed popełnieniem omyłki przy przepisywaniu programu. PAMIĘTAJ: ZAWSZE UŻYWAJ „EDYTORA BASICA"

```
NC  32000 REM EDYTOR BASICA
SE  32010 REM wersja 1.1 dla "Bajtka"
JB  32020 CLR :DIM LINIA$(120):CLOSE #2:CL
    OSE #3
MN  32030 OPEN #2,4,0,"E:":OPEN #3,5,0,"E:
    "
ZR  32040 ? CHR$(125):POSITION 11,1:? "ED
    YTOR BASICA "
YB  32050 TRAP 32040:POSITION 2,3:? "Wpisz
     linie programu"
KH  32060 POSITION 1,4:? " ":INPUT #2;LINI
    A$:IF LINIA$="" THEN POSITION 2,4:LIST
     B:GOTO 32060
XL  32070 IF LINIA$(1,1)="*" THEN B=VAL(LI
    NIA$(2,LEN(LINIA$))):POSITION 2,4:LIST
     B:GOTO 32060
TH  32080 POSITION 2,10:? "CONT"
BI  32090 B=VAL(LINIA$):POSITION 1,3:? " "
    ;
NY  32100 POKE 842,13:STOP
CN  32110 POKE 842,12
KI  32120 ? CHR$(125):POSITION 11,1:? "ED
    YTOR BASICA ":POSITION 2,15:LIST B
BZ  32130 C=0:ODP=C
LL  32140 POSITION 2,16:INPUT #3;LINIA$:IF
     LINIA$="" THEN ? "LINIA ";B;" USUNIET
    A":GOTO 32050
GU  32150 FOR D=1 TO LEN(LINIA$):C=C+1:ODP
    =ODP+(C*ASC(LINIA$(D,D))):NEXT D
SR  32160 KOD=INT(ODP/676)
BX  32170 KOD=ODP-KOD*676
BM  32180 KODS=INT(KOD/26)
VF  32190 KODM=KOD-(KODS*26)+193
NF  32200 KODS=KODS+193
UW  32210 POSITION 0,16:? CHR$(KODS);CHR$(
    KODM)
HN  32220 POSITION 2,13:? "Jezeli kod sie
    nie zgadza, nacisnij  RETURN  i popr
    aw linie.":GOTO 32050
```

WPISYWANIE ZNAKÓW SPECJALNYCH

Część programów publikowana w naszym piśmie jest drukowana w trybie graficznym.

Jest to spowodowane występowaniem w tych programach znaków, które są stosowane tylko w komputerach Atari i nie mogą być przedstawione przez drukarkę w trybie znakowym.

W trybie graficznym znaki znajdujące się obok siebie są drukowane bez odstępów (jak na ekranie monitora), lecz między poszczególnymi wierszami pozostaje niewielki odstęp (inaczej niż na ekranie). Poniżej pokazany jest wygląd wszystkich znaków uwidocznionych na klawiaturze.

Zestaw znaków Atari zawiera ponadto znaki specjalne, które wymagają specjalnego wpisywania. Wygląd tych znaków oraz klawisze, które służą do ich uzyskania, są pokazane w tabelkach umieszczonych obok. Zawsze, gdy są używane klawisze ‹CONTROL› (w tabelach CTRL") i ›SHIFT‹, trzeba trzymać je wciśnięte podczas naciskania następnego klawisza. Gdy konieczne jest użycie klawisza ›ESC‹, należy go nacisnąć tylko jeden raz i zwolnić przed naciśnięciem następnego klawisza. Znaki pokazane w tabeli „INVERSE" są negatywami znaków z tabeli „NORMALNIE" Przełączenie na znaki w negatywie i odwrotnie następuje po naciśnięciu klawisza z symbolem przekreślonego prostokąta (tzw. klawisz Atari). Przełączanie na pisanie w negatywie (lub odwrotnie) nie jest konieczne dla uzyskania znaków, które są uzyskiwane przy użyciu ›ESC‹. Są one zawsze wyświetlane w sposób pokazany w tabelach.

Błędy najczęściej występujące przy przepisywaniu programów są powodowane przez wpisanie dużej litery zamiast małej — dotyczy to szczególnie liter P, X i O. Ponadto łatwo pomylić literę O z cyfrą 0, kropkę (.) z przecinkiem (,) oraz średnik (;) z dwukropkiem (:). Podobnie, niektóre znaki specjalne mają wygląd bardzo zbliżony do zwykłych znaków alfanumerycznych. Na wydruku rozróżnić je można tylko według grubości linii. Dla porównania i zapamiętania znaki są pokazane poniżej.

SPECJALNE

CTRL F	
CTRL G	
CTRL N	
CTRL R	
CTRL S	

NORMALNIE

znak /	
SHIFT +	
SHIFT -	
znak -	
znak +	

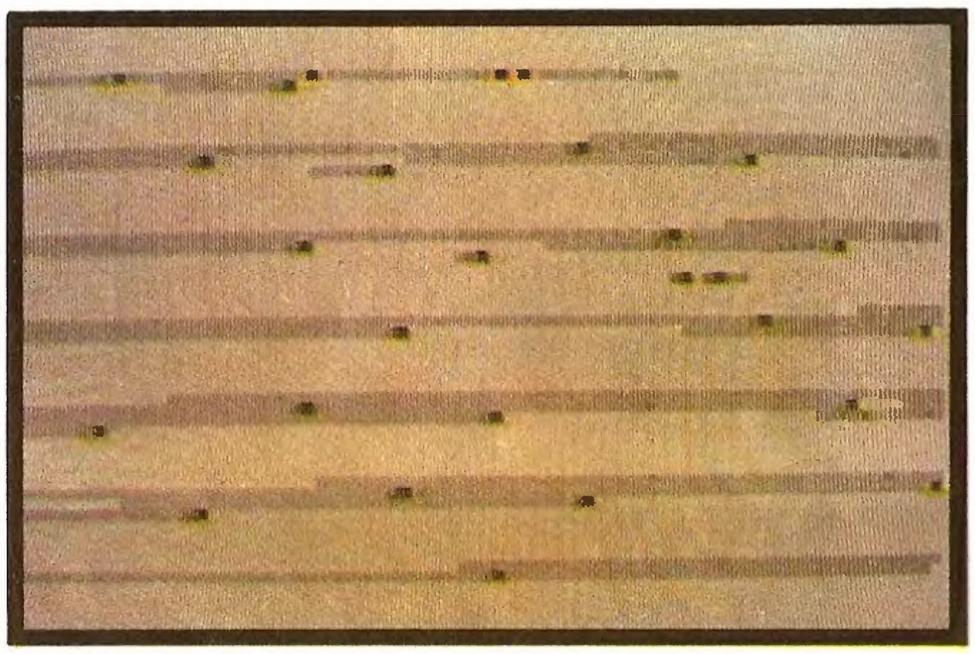
NORMALNIE

CTRL ,	CTRL S
CTRL A	CTRL T
CTRL B	CTRL U
CTRL C	CTRL V
CTRL D	CTRL W
CTRL E	CTRL X
CTRL F	CTRL Y
CTRL G	CTRL Z
CTRL H	ESC ESC
CTRL I	ESC CTRL -
CTRL J	ESC CTRL =
CTRL K	ESC CTRL +
CTRL L	ESC CTRL *
CTRL M	CTRL .
CTRL N	CTRL ;
CTRL O	SHIFT =
CTRL P	ESC CTRL CLEAR
CTRL Q	ESC DELETE
CTRL R	ESC TAB

INVERSE

CTRL ,	CTRL S
CTRL A	CTRL T
CTRL B	CTRL U
CTRL C	CTRL V
CTRL D	CTRL W
CTRL E	CTRL X
CTRL F	CTRL Y
CTRL G	CTRL Z
CTRL H	ESC SHIFT DELETE
CTRL I	ESC SHIFT INSERT
CTRL J	ESC CTRL TAB
CTRL K	ESC SHIFT TAB
CTRL L	CTRL .
CTRL M	CTRL ;
CTRL N	SHIFT =
CTRL O	ESC CTRL 2
CTRL P	ESC CTRL DELETE
CTRL Q	ESC CTRL INSERT
CTRL R	

GRA: WOJNY RDZENIOWE



Z pewnością wielu z Was oglądało film „Tron". Jego tematem są zmagania dobrych i złych programów we wnętrzu komputera. Film jest wprawdzie dość stary, lecz współczesną jego wersją są w rzeczywistości boje toczone pomiędzy wirusami i programami przeznaczonymi do ich niszczenia.

Tak atrakcyjny temat już dawno doczekał się komputerowej realizacji w postaci gry „Wojny rdzeniowe" najczęściej znanej pod angielskim tytułem „Core Wars". Niestety, nie spotkałem dotychczas wersji tej gry przeznaczonej dla ośmiobitowych komputerów Atari. Pora nadrobić więc to zaniedbanie.

ZASADY

W „Wojnach rdzeniowych" rozgrywka toczy się między dwoma programami napisanymi przez zawodników. Najciekawszą i zarazem najtrudniejszą fazą gry jest ułożenie programu, który będzie w stanie pokonać przeciwnika. Kłopot polega głównie na tym, że nie wiadomo z góry, kto tym przeciwnikiem będzie. Po wprowadzeniu walczących programów pozostaje tylko obserwacja toczącej się bitwy i oczekiwanie na zwycięstwo jednego z graczy.

Polem walki programów jest obszar pamięci zamknięty w pętlę bez końca. Przekroczenie przez jeden z programów końca tego obszaru powoduje pojawienie się go na początku i odwrotnie. Walczące programy są umieszczone losowo w pamięci komputera i na przemian wykonywane. Podczas wykonywania programy mogą przemieszczać się w pamięci i dzielić. Jeśli w przypadającej na niego kolejności program nie może być prawidłowo wykonany, to zostaje zniszczony. Gracz, który stracił wszystkie programy, przegrywa.

Programy wojenne są pisane w specjalnym języku zwanym „Redcode". Do ich tworzenia można wykorzystać dowolny edytor tekstowy (bardzo dobry jest edytor **Action!**). Program wojenny musi być zapisany w postaci pliku na kasecie lub dyskietce, skąd jest odczytywany przez grę.

REDCODE

Język „Redcode" w wersji dla Atari XL/XE, której dotyczy dalszy opis, jest nieco uproszczony w stosunku do standardu, co zostało wymuszone niewielką pojemnością pamięci komputera. Język ten zawiera dziesięć rozkazów pokazanych w tabeli i opisanych poniżej.

DAT — rozkaz zawierający wyłącznie dane i niemożliwy do wykonania (próba taka niszczy program);
MOV — przepisz zawartość komórki A do komórki B, a jeśli A jest daną, to utwórz w komórce B rozkaz DAT;
ADD — dodaj argumenty A i B, a wynik umieść w komórce B;
SUB — odejmij argumenty A i B, a wynik umieść w komórce B;
JMP — skocz do komórki określonej przez argument;
JMZ — skocz do komórki określonej przez argument A, gdy zawartość komórki B jest równa zero;
JMN — skocz do komórki określonej przez argument A, gdy zawartość komórki B jest różna od zera;
DJN — zmniejsz o jeden argument B i skocz do komórki określonej przez argument A, gdy wynik jest różny od zera;
CMP — porównaj zawartości A i B, a jeśli są różne, to opuść (przeskocz) następny rozkaz;
SPL — podziel się na części: jedną, wykonywaną od następnego rozkazu i drugą, wykonywaną od rozkazu wskazanego przez argument.

Argumentami rozkazów mogą być wyłącznie liczby określające dane i adresy. Adresy są zawsze podawane względnie, czyli licząc od rozkazu, który zawiera dany adres. Dane, czyli liczby używane bezpośrednio, a nie jako adresy oznacza się znakiem „#". Na przykład:

MOV 10 20

oznacza przepisanie zawartości komórki odległej od tego rozkazu o 10 do komórki, której adres znajduje się w komórce odległej o 20, a

MOV #10 20

oznacza umieszczenie w komórce, której adres jest odległy o 20, liczby 10.

Ponadto dopuszczone jest stosowanie w argumentach znaków „<", „>", i „@". Oznaczają one dodanie do wartości argumentu pobranego ze wskazanej komórki odpowiednio wartości -1, 1 lub 0. Rozkaz

MOV 10 <20

oznacza więc przepisanie zawartości komórki odległej o 10 do komórki, której adresem jest zmniejszona o jeden zawartość komórki odległej o 20.

W odróżnieniu od innych wersji „Redcode", na Atari nie można stosować etykiet i wszystkie adresy i dane muszą być podawane bezpośrednio liczbami. Oczywiście użycie liczby ujemnej nakazuje odliczanie wstecz od rozkazu, który ją zawiera.

GRA

Działanie programów wojennych jest nadzorowane przez właściwy program gry o nazwie MARS *(Memory Array Redcode Simulator)*. Program ten jest przedstawiony na listingu 1. Dla ułatwienia analizy w programie znajdują się liczne komentarze oddzielające poszczególne fragmenty. Listing ten należy uważnie przepisać (przy użyciu „Edytora Basica") pomijając wszystkie wiersze zawierające komentarze (rozpoczynające się instrukcją REM). Program z komentarzami nie będzie poprawnie działał, gdyż pozostały obszar wolnej pamięci komputera będzie niewystarczający.

W celu rozpoczęcia gry należy uruchomić program MARS. Następnie podaje się nazwy plików zawierających wybrane programy wojenne. Zostaną one odczytane i rozpoczną walkę. Walka ta toczy się, aż do skutku, czyli do chwili, gdy jeden z programów zostanie całkowicie zniszczony. Jeżeli trzeba przerwać grę, wystarczy nacisnąć klawisz ‹BREAK›.

Program przystosowany jest do współpracy zarówno z magnetofonem, jak i ze stacją dysków. Przy korzystaniu ze stacji podaje się pełną specyfikację pliku zawierającego program wojenny, np. **"D:WORM.CW"**. Jeżeli program wojenny będzie odczytywany z magnetofonu, to należy ustawić taśmę na jego początku i wpisać tylko specyfikację urządzenia **"C:"**.

PROGRAMY WOJENNE

Ponieważ na początku ułożenie własnego programu wojennego może sprawiać pewną trudność, to kilka najprostszych (co nie znaczy najsłabszych) zostało pokazane na listingach 2—9.

Najprostszym programem wojennym jest **SKOCZEK** (listing 2), który składa się tylko z jednego rozkazu. Kopiuje on samego siebie do następnej komórki pamięci i w ten sposób systematycznie „zadeptuje" jej zawartość.

BLIŹNIAKI (listing 3) wędrują po pamięci kopiując się do obszaru położonego o 100 komórek dalej.

KARZEŁ (listing 4) próbuje „upolować" przeciwnika strzelając w co piątą komórkę pamięci.

Trudnym przeciwnikiem jest również **ZMIENIACZ** (listing 5), który dzieli się podczas walki na części o odmiennym działaniu.

Podobnie **KOPIER** (listing 6) dzieli się na części, które rozrzucane są po całym obszarze pamięci.

Rozkazy Redcode

symbol	rozkaz	argumenty	operacja
0	DAT	A	dane
1	MOV	A B	przepisz
2	ADD	A B	dodaj
3	SUB	A B	odejmij
4	JMP	A	skocz
5	JMZ	A B	skocz, gdy zero
6	JMN	A B	skocz, gdy nie zero
7	DJN	A B	zmniejsz i skocz, gdy nie zero
8	CMP	A B	porównaj
9	SPL	A	podziel się

ROBAK (listing 7) stanowi groźnego, choć powolnego przeciwnika. Posuwa się on powoli do przodu kontrolując przy tym swój „ogon", dzięki czemu unika „zadeptania" od tyłu.

SKOCZEK ZBROJNY (listing 8) jest odmianą SKOCZKA, która dzieli się co 200 kolejek tworząc nowego SKOCZKA.

Analogicznie mnożą się **BLIŹNIAKI ZBROJNE** (listing 9), lecz efektem podziału są rosnące zastępy BLIŹNIAKÓW.

Życzę wszystkim graczom powodzenia w „Wojnach rdzeniowych" i proszę o nadsyłanie własnych programów wojennych. Najciekawsze z nich zostaną opublikowane w „Moim Atari".

Wojciech ZIENTARA

LISTING 1

```
MQ    10 REM *      WOJNY RDZENIOWE     *
NV    20 REM *    Wojciech Zientara     *
GB    30 REM *  (c) 1991, Sp. Bajtek    *
WE    40 REM -------------------------
KY    50 REM **    Inicjowanie    **
WG    60 REM -------------------------
PN    70 GRAPHICS 0:S=3840
DQ    80 DIM A$(S),B$(S),C$(S),D$(S),E$(S),F
      $(S),G$(S)
IR    90 DIM PC(100,1),PE(1),PM(1),S(1),C(1)
      ,Z$(20),Y$(3),O$(10),I$(1),M$(1),N$(1)
AN    100 A$="O":A$(S)="O":A$(2)=A$:B$=A$:C$
      =A$:D$=A$:E$=A$:F$=A$:G$=A$
FX    110 REM -------------------------
LT    120 REM **    odczyt programow    **
GB    130 REM -------------------------
TE    140 S(0)=1:S(1)=1420+INT(RND(0)*1001):
      C(0)=S(0):C(1)=S(1):FATAL=0
CE    150 FOR I=0 TO 1:TRAP 2140
OT    160 ? "PROGRAM WOJENNY ";I+1;": ";
IC    170 INPUT Z$:Q=1:OPEN #1,4,0,Z$:TRAP 7
      70
DI    180 INPUT #1,Z$:MAX=LEN(Z$):J=1
GN    190 REM -------------------------
GJ    200 REM * Okreslenie kodu rozkazu *
FY    210 REM -------------------------
YD    220 GOSUB 2180:IF EROR THEN 2090
VS    230 RESTORE :L=0
EH    240 READ Y$:IF Z$(J,J+2)=Y$ THEN 270
NN    250 L=L+1:IF L<10 THEN 240
OH    260 GOTO 2090
PU    270 O$=STR$(L):J=J+3:IF J>MAX THEN 209
      0
BN    280 DATA DAT,MOV,ADD,SUB,JMP,JMZ,JMN,D
      JN,CMP,SPL
GO    290 REM -------------------------
HX    300 REM **    Tryb operandu A    **
FZ    310 REM -------------------------
CI    320 IF O$="O" THEN 510
YG    330 GOSUB 2180:IF EROR THEN 2090
XS    340 I$=Z$(J,J):IF I$="#" OR I$="@" OR
      I$="<" OR I$=">" THEN M$=I$:GOTO 370
CK    350 IF I$="+" OR I$="-" OR ASC(I$)>47
      AND ASC(I$)<58 THEN M$=" ":GOTO 420
OI    360 GOTO 2090
XB    370 J=J+1:IF J>MAX THEN 2090
GN    380 REM -------------------------
TM    390 REM **    Operand A    **
FY    400 REM -------------------------
NG    410 I$=Z$(J,J)
SB    420 IF I$="+" OR I$="-" OR ASC(I$)>47
      AND ASC(I$)<58 THEN 440
OD    430 GOTO 2090
OE    440 IF O$<>"4" AND O$<>"9" THEN 460
ID    450 OPA=VAL(Z$(J)):GOTO 650
WE    460 FOR L=J TO MAX:IF Z$(L,L)<>" " THE
      N NEXT L:L=O:GOTO 2090
BC    470 POP :OPA=VAL(Z$(J,L-1)):J=L
GO    480 REM -------------------------
JS    490 REM **    Tryb operandu B    **
FZ    500 REM -------------------------
YE    510 GOSUB 2180:IF EROR THEN 2090
VG    520 I$=Z$(J,J):IF I$="@" OR I$="<" OR
      I$=">" THEN N$=I$:GOTO 550
AS    530 IF I$="+" OR I$="-" OR ASC(I$)>47
      AND ASC(I$)<58 THEN N$=" ":GOTO 590
OG    540 GOTO 2090
WZ    550 J=J+1:IF J>MAX THEN 2090
GL    560 REM -------------------------
UG    570 REM **    Operand B    **
GP    580 REM -------------------------
MJ    590 I$=Z$(J,J):IF I$="+" OR I$="-" OR
      ASC(I$)>47 AND ASC(I$)<58 THEN GOTO 61
      0
NZ    600 GOTO 2090
QC    610 OPB=VAL(Z$(J))
GE    620 REM -------------------------
BC    630 REM * Przepisanie do pamieci *
GI    640 REM -------------------------
OC    650 K=C(I):V=INT((K-1)/80):H=K-INT((K-
      1)/80)*80
BW    660 A$(K,K)=O$:B$(K,K)=M$:IF OPA<0 THE
      N OPA=32767+ABS(OPA)
JY    670 C$(K,K)=CHR$(OPA-INT(OPA/256)*256)
      :D$(K,K)=CHR$(INT(OPA/256)):E$(K,K)=N$
MC    680 IF OPB<0 THEN OPB=32767+ABS(OPB)
JI    690 F$(K,K)=CHR$(OPB-INT(OPB/256)*256)
      :G$(K,K)=CHR$(INT(OPB/256))
GB    700 REM -------------------------
IH    710 REM **  Nastepny wiersz  **
GF    720 REM -------------------------
LM    730 C(I)=C(I)+1:Q=Q+1:GOTO 180
GJ    740 REM -------------------------
EI    750 REM **  Koniec odczytu  **
GN    760 REM -------------------------
CL    770 CLOSE #1:TRAP 40000
GN    780 NEXT I
ZC    790 IF FATAL THEN ? :? "BLAD W PROGRAM
      IE WOJENNYM - PROGRAM PRZERWANY !":END
GC    800 REM -------------------------
IJ    810 REM **  Program wlasciwy  **
GG    820 REM -------------------------
RF    830 GRAPHICS 21:SETCOLOR 4,0,12:SETCOL
      OR 0,3,4:SETCOLOR 1,12,6:SETCOLOR 2,0,
      10
XM    840 PE(0)=1:PM(0)=1:PE(1)=1:PM(1)=1:PC
      (PE(0),0)=S(0):PC(PE(1),1)=S(1)
GM    850 REM -------------------------
AL    860 POKE 77,0:FOR P=0 TO 1:IF PM(P)=0
      THEN GRAPHICS 53:? :? "PROGRAM ";P+1;"
       PRZEGRAL !":END
TY    870 MEM=PC(PE(P),P)
GY    880 GOSUB 1110:INS=0:ERA=0
CO    890 GOSUB 1190:MEM=MEM-(MEM>S-1)*S:MEM
      =MEM+(MEM=0):PC(PE(P),P)=MEM
CN    900 IF INS THEN GOSUB 980
XB    910 ON ERA+1 GOSUB 1070,1030
ZZ    920 PE(P)=PE(P)+1:IF PE(P)>PM(P) THEN
      PE(P)=1
IX    930 NEXT P
RC    940 GOTO 860
GN    950 REM -------------------------
CW    960 REM **    Nowy adres    **
GR    970 REM -------------------------
VI    980 IF PM(P)=100 THEN RETURN
GN    990 FOR I=PM(P) TO PE(P) STEP -1:PC(I+
      1,P)=PC(I,P):NEXT I:PM(P)=PM(P)+1:PC(P
      E(P),P)=NMEM:PE(P)=PE(P)+1:RETURN
RL    1000 REM -------------------------
OA    1010 REM **  Usuniecie adresu  **
RR    1020 REM -------------------------
QH    1030 FOR I=PE(P)+1 TO PM(P):PC(I-1,P)=
      PC(I,P):NEXT I:PM(P)=PM(P)-1:PE(P)=PE(
      P)-1:RETURN
RX    1040 REM -------------------------
AV    1050 REM **  Ustawienie punktu  **
SD    1060 REM -------------------------
KR    1070 COLOR P+1:V=INT((MEM-1)/80):H=MEM
      -INT((MEM-1)/80)*80-1:PLOT H,V:RETURN
SJ    1080 REM -------------------------
XH    1090 REM **  Skasowanie punktu  **
RN    1100 REM -------------------------
AP    1110 V=INT((MEM-1)/80):H=MEM-INT((MEM-
      1)/80)*80-1:LOCATE H,V,Z:COLOR (Z<>O)*
      3:PLOT H,V:RETURN
RZ    1140 REM -------------------------
EC    1150 REM ** Interpreter rozkazow **
SF    1160 REM -------------------------
FF    1170 REM **    Operand A    **
SL    1180 REM -------------------------
YR    1190 I$=B$(MEM,MEM):IF I$=" " OR I$="@
      " THEN I=O
RN    1200 IF I$="<" THEN I=-1
CY    1210 IF I$=">" THEN I=1
CF    1220 DA=ASC(C$(MEM,MEM))+ASC(D$(MEM,ME
      M))*256:IF DA>32767 THEN DA=32767-DA
FG    1230 IF I$="#" THEN 1300
CE    1240 DA=DA+MEM:DA=DA-INT(DA/S)*S:DA=DA
      +(DA=0)
XN    1250 OPA=ASC(F$(DA,DA))+ASC(G$(DA,DA))
      *256+1:IF OPA>32767 THEN OPA=32767-OPA
HQ    1260 IF I$=" " THEN 1340
RR    1270 OP=OPA:IF OPA<0 THEN OP=32767+ABS
      (OPA)
VY    1280 F$(DA,DA)=CHR$(OP-INT(OP/256)*256
      ):G$(DA,DA)=CHR$(INT(OP/256))
WU    1290 DA=OPA:I$=" ":GOTO 1240
DL    1300 OPA=DA
RU    1310 REM -------------------------
FR    1320 REM **    Operand B    **
SA    1330 REM -------------------------
ZH    1340 I$=E$(MEM,MEM):IF I$=" " OR I$="@
      " THEN I=0
SE    1350 IF I$="<" THEN I=-1
DP    1360 IF I$=">" THEN I=1
PK    1370 DB=ASC(F$(MEM,MEM))+ASC(G$(MEM,ME
      M))*256:IF DB>32767 THEN DB=32767-DB
KG    1380 DB=DB+MEM:DB=DB-INT(DB/S)*S:DB=DB
      +(DB=0)
JY    1390 OPB=ASC(F$(DB,DB))+ASC(G$(DB,DB))
      *256+1:IF OPB>32767 THEN OPB=32767-OPB
JS    1400 IF I$=" " THEN 1460
TW    1410 OP=OPB:IF OPB<0 THEN OP=32767+ABS
      (OPB)
NU    1420 GOSUB 2050:DB=OPB:I$=" ":GOTO 138
      0
SC    1430 REM -------------------------
IC    1440 REM **  Wykonanie rozkazu  **
SI    1450 REM -------------------------
XT    1460 CMD=ASC(A$(MEM,MEM))-47
XQ    1470 ON CMD GOSUB 1520,1560,1650,1690,
      1750,1790,1850,1910,1960,2010
BF    1480 RETURN
SU    1490 REM -------------------------
SP    1500 REM **  Rozkaz 0: DAT  **
RY    1510 REM -------------------------
ZW    1520 ERA=1:RETURN
SE    1530 REM -------------------------
RQ    1540 REM **  Rozkaz 1: MOV  **
SK    1550 REM -------------------------
AC    1560 IF B$(MEM,MEM)="#" THEN 1590
QC    1570 A$(DB,DB)=A$(DA,DA):B$(DB,DB)=B$(
      DA,DA):C$(DB,DB)=C$(DA,DA):D$(DB,DB)=D
      $(DA,DA)
XP    1580 E$(DB,DB)=E$(DA,DA):F$(DB,DB)=F$(
      DA,DA):G$(DB,DB)=G$(DA,DA):GOTO 1610
SD    1590 OP=OPA:IF OPA<0 THEN OP=32767+ABS
      (OPA)
ZZ    1600 GOSUB 2050
AA    1610 MEM=MEM+1:RETURN
SD    1620 REM -------------------------
ET    1630 REM **  Rozkaz 2: ADD  **
SJ    1640 REM -------------------------
IJ    1650 OPB=OPB+OPA:GOTO 1700
SP    1660 REM -------------------------
KZ    1670 REM **  Rozkaz 3: SUB  **
SV    1680 REM -------------------------
PG    1690 OPB=OPB-OPA
TZ    1700 OP=OPB:IF OPB<0 THEN OP=32767+ABS
      (OPB)
XR    1710 GOSUB 2050:MEM=MEM+1:RETURN
SF    1720 REM -------------------------
JK    1730 REM **  Rozkaz 4: JMP  **
SL    1740 REM -------------------------
FX    1750 MEM=DA:RETURN
SR    1760 REM -------------------------
UR    1770 REM **  Rozkaz 5: JMZ  **
SX    1780 REM -------------------------
RY    1790 IF OPB<>0 THEN MEM=MEM+1
UW    1800 IF OPB=0 THEN MEM=DA
AS    1810 RETURN
SH    1820 REM -------------------------
JC    1830 REM **  Rozkaz 6: JMN  **
SN    1840 REM -------------------------
AA    1850 IF OPB=0 THEN MEM=MEM+1
GC    1860 IF OPB<>0 THEN MEM=DA
BK    1870 RETURN
SZ    1880 REM -------------------------
CE    1890 REM **  Rozkaz 7: DJN  **
SD    1900 REM -------------------------
PO    1910 OPB=OPB-1:OP=OPB:IF OPB<0 THEN OP
      =32767+ABS(OPB)
RJ    1920 GOSUB 2050:GOSUB 1850:RETURN
SM    1930 REM -------------------------
GL    1940 REM **  Rozkaz 8: CMP  **
SS    1950 REM -------------------------
NC    1960 IF OPA<>OPB THEN MEM=MEM+1
AY    1970 MEM=MEM+1:RETURN
TB    1980 REM -------------------------
VM    1990 REM **  Rozkaz 9: SPL  **
RM    2000 REM -------------------------
YG    2010 NMEM=DA:MEM=MEM+1:INS=1:RETURN
RS    2020 REM -------------------------
YR    2030 REM **  Zapis operandu  **
RY    2040 REM -------------------------
QA    2050 F$(DB,DB)=CHR$(OP-INT(OP/256)*256
      ):G$(DB,DB)=CHR$(INT(OP/256)):RETURN
SE    2060 REM -------------------------
MY    2070 REM **  Blad w programie  **
SK    2080 REM -------------------------
ZS    2090 ? :? "BLAD W PROGRAMIE W WIERSZU
      ";Q
EK    2100 FATAL=1:Q=Q+1:GOTO 180
RR    2110 REM -------------------------
NU    2120 REM **    Blad odczytu    **
RX    2130 REM -------------------------
HW    2140 ? :? "BLAD ODCZYTU Z DYSKIETKI -
      PROGRAM PRZERWANY !":END
SD    2150 REM -------------------------
EA    2160 REM **  Przeskok spacji  **
SJ    2170 REM -------------------------
HA    2180 IF Z$(J,J)<>" " THEN EROR=O:RETUR
      N
CD    2190 J=J+1:IF J<=MAX THEN 2180
FB    2200 EROR=1:RETURN
PO    2210 REM ****
```

LISTING 2

```
MOV 0 1
```

GRY

LISTING 3

```
JMP 3
DAT 0
DAT 99
MOV -2 -1
CMP #9 -3
JMP 4
ADD #1 -5
ADD #1 -5
JMP -5
MOV #99 93
JMP 93
```

LISTING 4

```
JMP 2
DAT 0
ADD #5 -1
MOV #0 -2
JMP -2
```

LISTING 5

```
JMP 4
MOV #0 -1
JMP -1
DAT 9
SPL -2
SPL 4
ADD #-16 -3
MOV #0 @-4
JMP -4
SPL 2
JMP -1
MOV 0 1
```

LISTING 6

```
SPL 4
MOV #0 -2
JMP -1
MOV 0 1
MOV -1 @4
SPL <3
ADD #653 2
JMP -3
DAT 833
```

LISTING 7

```
JMP 2
DAT 5
SPL 4
DJN 2 -2
JMP -1
MOV #0 1
SPL 0
MOV 0 1
```

LISTING 8

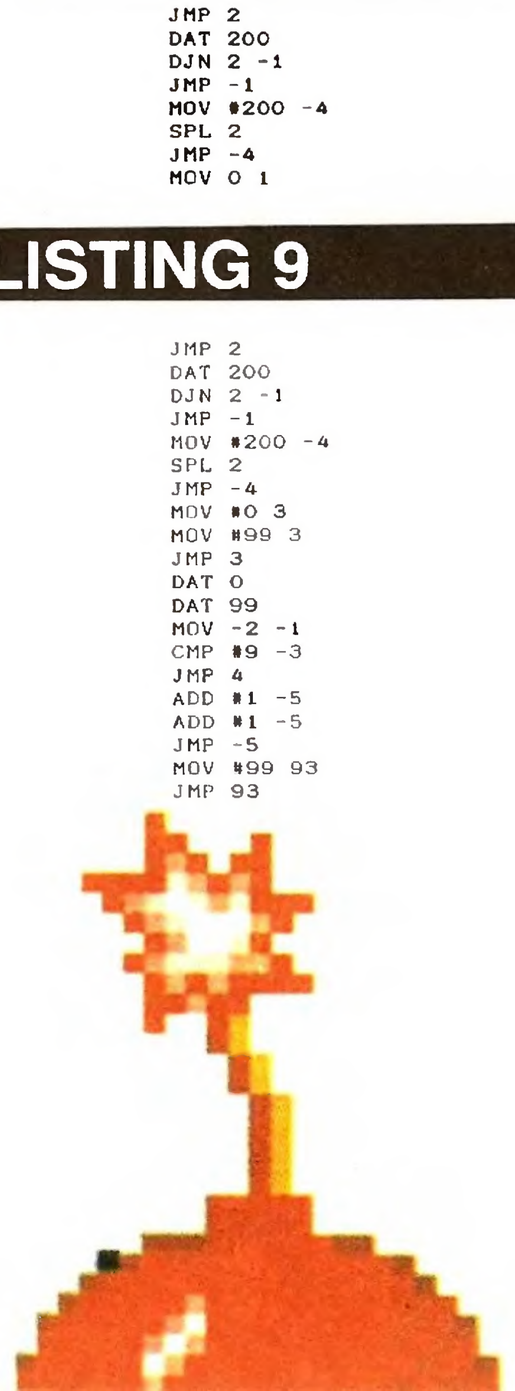
```
JMP 2
DAT 200
DJN 2 -1
JMP -1
MOV #200 -4
SPL 2
JMP -4
MOV 0 1
```

LISTING 9

```
JMP 2
DAT 200
DJN 2 -1
JMP -1
MOV #200 -4
SPL 2
JMP -4
MOV #0 3
MOV #99 3
JMP 3
DAT 0
DAT 99
MOV -2 -1
CMP #9 -3
JMP 4
ADD #1 -5
ADD #1 -5
JMP -5
MOV #99 93
JMP 93
```

WOJNY

RDZENIOWE



KATALOG

DYSKIETKI W BASICU

Jestem już od wielu lat użytkownikiem komputera Atari 800XL. Podczas pracy w Atari Basic stwierdziłem irytujące niedostosowanie języka do współpracy ze stacją dysków. Brak w nim przede wszystkim bardzo użytecznej instrukcji DIR. Poniższy program poprawia ten mankament, przypisując standardowej instrukcji DOS nowy sposób działania.

Program jest przeznaczony do współpracy z DOS 2.5 i musi być zapisany na dyskietce pod nazwą AUTORUN.SYS. Składa się on z dwóch części: pierwsza rezyduje stale w pamięci i zajmuje lokacje bezpośrednio za obszarem DOS (od $1C60 do $1DDE), zaś druga jest potrzebna tylko do instalacji programu, a potem ulega skasowaniu (zajmuje ona obszar od $7000 do $7065).

WPROWADZENIE PROGRAMU

Do utworzenia właściwego programu służy zamieszczony obok pomocniczy program w Basicu. Zawiera on w instrukcjach DATA dane niezbędne do stworzenia pliku AUTORUN.SYS. Dla uzyskania takiego pliku musisz wykonać następujące czynności:

1. Przygotuj nową dyskietkę, która będzie zawierała pliki DOS.SYS i DUP.SYS (zapisane przy użyciu funkcji „H" z menu DOS).
2. Przepisz zamieszczony program kontrolując jego poprawność przy pomocy „Edytora Basica"
3. Zapisz (na wszelki wypadek) przepisany program przy użyciu instrukcji SAVE "D:CREAUTO".
4. Włóż do stacji dyskietkę przygotowaną według wskazówek zawartych w punkcie 1.
5. Uruchom przepisany program instrukcją RUN i chwilę poczekaj.
6. Jeżeli pojawi się komunikat "BLAD", to sprawdź zgodność przepisanego programu z wydrukowanym listingiem i wróć do punktu 3.
7. Po ukazaniu się komunikatu "READY" wyłącz komputer i uruchom go ponownie nie zmieniając dyskietki w stacji dysków. Na ekranie pojawi się napis:
   DOS −> DIR D:*.*
   READY

DZIAŁANIE PROGRAMU

Program jest zapisany na dyskietce pod nazwą AUTORUN.SYS, jest zatem automatycznie uruchamiany podczas inicjowania systemu. Najpierw sprawdzana jest obecność Basica. Jeżeli Basic jest odłączony, to program nie zostaje zainstalowany. Po zainstalowaniu programu i wyświetleniu komunikatu

DOS −> DIR D:*.*

każdorazowe wykonanie instrukcji DOS powoduje odczytanie katalogu dyskietki znajdującej się w stacji numer 1 i wyświetlenie go na ekranie. W celu sprawdzenia obecności zmienionej instrukcji DOS wystarczy nacisnąć klawisz ›RESET‹. Istnienie procedury zostanie ponownie zasygnalizowane komunikatem

OCHRONA PRZED ZAPISEM W STACJACH

LDW i CA

Każdy, kto miał kiedykolwiek do czynienia z dyskietkami 5,25" zapewne wie, jak są one chronione przed niepożądanym zapisem.

Otóż w napędzie odczytującym i zapisującym takie dyskietki jest zainstalowana dioda emitująca promienie podczerwone i fototranzystor, który je odbiera. Elementy te są umieszczone w taki sposób, że dyskietka wsunięta w kieszeń napędu przesłania odbiornik, co powoduje brak zezwolenia na zapis. Jeżeli natomiast w dyskietce jest odpowiednie wycięcie, to promienie wysyłane przez diodę dochodzą do fototranzystora i dzięki temu zaczyna on przewodzić, co jest równoznaczne z zezwoleniem na zapis lub formatowanie.

Większość dyskietek dostępnych w handlu ma takie wycięcie wykonane fabrycznie, ale tylko z jednej strony. Co zatem zrobić, aby sformatować lub zapisać dyskietkę na drugiej stronie, jeżeli mamy jednostronną stację typu LDW SUPER 2000 lub CALIFORNIA ACCES 2001? Do wyboru mamy dwie możliwości: wykonanie odpowiedniego wycięcia w dyskietce, co w przypadku braku specjalnego narzędzia nie jest łatwe i z pewnością nie będzie zbyt estetyczne, lub spowodowanie, że stacji dysków będzie się wydawało, iż takie wycięcie istnieje. Najprostszą metodą zrealizowania drugiego wariantu byłoby zwarcie fototranzystora czyli stworzenie takiej samej sytuacji jaka występuje, gdy jest on w stanie przewodzenia. Niestety sposób ten, skuteczny w przypadku stacji typu 1050, nie może być zastosowany w LDW i CA, albowiem dzięki czujnikowi, o którym mowa, stacje te dowiadują się o ewentualnej zmianie dyskietki i przy próbie odczytu ustawiają się w odpowiedniej gęstości. Mając to na uwadze proponuję użytkownikom LDW i CA wykonanie drobnej przeróbki, która zmieni sposób ochrony dyskietki

DOS -> DIR D:*.*

Można odnieść wrażenie, że straciliśmy możliwość przejścia do DOS-u bez ponownej inicjacji systemu. Wystarczy jednak nacisnąć równocześnie klawisze ‹SELECT› i ‹RESET›. Jeżeli na dyskietce lub w ramdysku znajduje się plik DUP.SYS, to zgłosi się menu DOS. Jeżeli istnieje również plik MEM.SAV, to zachowany zostanie zarówno aktualnie opracowywany program w Basicu, jak i nasza dodatkowa procedura realizująca instrukcję DOS. Po powrocie do Basica nic nie ulegnie zmianie. Jeżeli jednak dyskietka lub ramdysk nie zawiera pliku MEM.SAV albo plik ten jest zabezpieczony przed zapisem, wówczas naciśnięcie ‹SELECT›+‹RESET› spowoduje wywołanie menu DOS oraz skasowanie programu w Basicu i naszej procedury. Nie pojawi się wtedy na ekranie komunikat sygnalizujący zmianę działania instrukcji DOS. Ponowne uruchomienie tej procedury wymaga wyłączenia komputera i ponownego zainicjowania systemu.

```
YB   10 REM DOS -> DIR
FW   20 REM Jaroslaw Kulinski
MY   30 REM (c) 1990, Bajtek
BB   40 REM
UN   100 REM KONTROLA WPISANYCH INSTR.DATA
UZ   110 ? CHR$(125);"CZEKAJ"
DX   120 SUM=0
HD   130 FOR I=1 TO 465
NZ   140 READ VAR
SR   150 SUM=SUM+VAR
GD   160 NEXT I
LA   170 IF SUM<>45054 THEN ? CHR$(253);"BL
     AD":END
KF   180 REM UTWORZENIE ZBIORU AUTORUN.SYS
WM   190 OPEN #1,8,0,"D:AUTORUN.SYS"
CB   200 RESTORE
HA   210 FOR I=1 TO 465
FQ   220 READ A:PUT #1,A
FY   230 NEXT I
LG   240 CLOSE #1
OB   250 END
YP   1000 DATA 255,255,132,28,222,29,155,68
     ,79,83,32,45,62,32,68,73,82,32,68,58,4
     2,46,42,155,32
OC   1010 DATA 255,255,169,223,141,231,2,16
     9,29,141,232,2,169,98,133,10,169,29,13
     3,11,173,31,208,201,5
FM   1020 DATA 240,31,162,0,169,11,157,66,3
     ,169,132,157,68,3,169,28,157,69,3,169,
     18,157,72,3,169
DS   1030 DATA 0,157,73,3,32,86,228,96,162,
     112,169,12,157,66,3,32,86,228,162,112,
     169,3,157,66,3
XZ   1040 DATA 169,4,157,74,3,169,0,157,75,
     3,76,249,28,68,58,68,79,83,46,83,89,83
     ,0,169,239
FZ   1050 DATA 157,68,3,169,28,157,69,3,32,
     86,228,48,170,162,112,169,12,157,66,3,
     32,86,228,162,112
FR   1060 DATA 169,3,157,66,3,169,12,157,74
     ,3,169,0,157,75,3,76,48,29,68,58,77,69
     ,77,46,83
SU   1070 DATA 65,86,0,169,38,157,68,3,169,
     29,157,69,3,32,86,228,16,32,173,126,28
     ,133,10,173,127
XQ   1080 DATA 28,133,11,173,128,28,141,231
     ,2,173,129,28,141,232,2,173,130,28,133
     ,12,173,131,28,133,13
BL   1090 DATA 108,126,28,162,112,169,12,15
     7,66,3,32,86,228,162,112,169,3,157,66,
     3,169,6,157,74,3
KI   1100 DATA 169,0,157,75,3,76,134,29,68,
     58,42,46,42,0,169,128,157,68,3,169,29,
     157,69,3,32
TV   1110 DATA 86,228,162,112,169,5,157,66,
     3,169,108,157,68,3,169,28,157,69,3,169
     ,255,157,72,3,169
BK   1120 DATA 0,157,73,3,32,86,228,48,33,1
     62,0,169,9,157,66,3,169,108,157,68,3,1
     69,28,157,69
CN   1130 DATA 3,169,255,157,72,3,169,0,157
     ,73,3,32,86,228,76,147,29,162,112,169,
     12,157,66,3,32
XU   1140 DATA 86,228,96,0,112,101,112,173,
     1,211,201,253,208,94,165,12,141,151,28
     ,141,130,28,165,13,141
PL   1150 DATA 152,28,141,131,28,169,150,13
     3,12,169,28,133,13,173,231,2,141,128,2
     8,169,223,141,231,2,173
NC   1160 DATA 232,2,141,129,28,169,29,141,
     232,2,165,10,141,126,28,165,11,141,127
     ,28,169,98,133,10,169
VO   1170 DATA 29,133,11,162,0,169,11,157,6
     6,3,169,132,157,68,3,169,28,157,69,3,1
     69,18,157,72,3
NG   1180 DATA 169,0,157,73,3,32,86,228,96,
     226,2,227,2,0,112
```

UWAGI TECHNICZNE

Podczas instalowania procedury mają miejsce następujące działania:

1. Sprawdzenie obecności interpretera Basica. Jeżeli jest on odłączony, to procedura nie jest instalowana.
2. Wartość wektora DOSINI ($0C–$0D) zostaje zmieniona tak, aby wskazywać nasz program, a jego wartość oryginalna zostaje zapamiętana.
3. Zawartość rejestru MEMLO ($02E7–$02E8) zostaje zmieniona tak, aby zabezpieczyć obszar pamięci zajmowany przez procedurę przed zniszczeniem przez Basic.
4. Wartość wektora DOSVEC ($0A–$0B) zostaje zmieniona tak, aby wskazywała nową procedurę obsługi instrukcji DOS, a jego wartość oryginalna zostaje zapamiętana.
5. Na ekranie jest wyświetlany komunikat "DOS -> DIR D:*.*".
6. Sterowanie komputerem zostaje przekazane do systemu operacyjnego.

Wykonanie w Basicu instrukcji DOS powoduje realizację następujących czynności:

1. Zostaje zamknięty kanał IOCB numer 7.
2. Zostaje otwarty kanał IOCB numer 7 do odczytu katalogu dyskietki (odpowiednik instrukcji Basica OPEN #7,6,0,"D:*.*").
3. Kolejne rekordy są odczytywane w pętli i wyświetlane na ekranie, aż do wyczerpania pliku.
4. Zostaje zamknięty kanał IOCB numer 7.
5. Sterowanie komputerem zostaje przekazane do Basica.

Naciśnięcie klawisza ‹RESET› powoduje wywołanie standardowej procedury gorącego startu sterowanej przez system operacyjny. Po jej zakończeniu wykonywana jest inicjacja naszej procedury, która obejmuje:

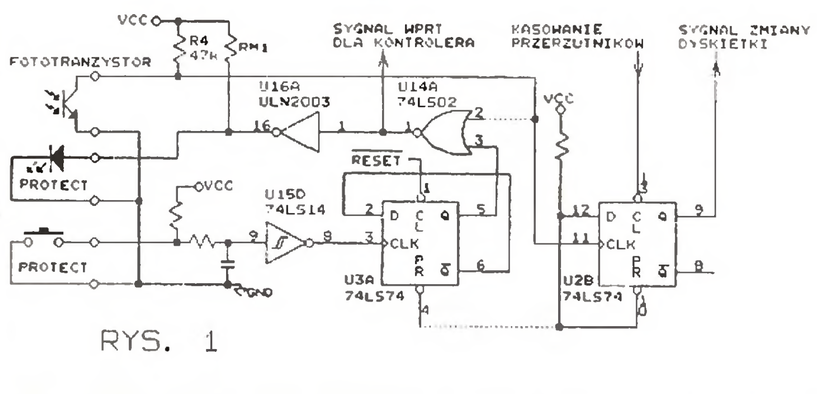
1. Inicjowanie DOS dzięki zapamiętanej pierwotnej wartości wektora DOSINI.
2. Ponowne zabezpieczenie obszaru pamięci zajętego przez procedurę.
3. Sprawdzenie stanu klawisza ‹SELECT›.
4. Jeżeli ‹SELECT› nie jest wciśnięty, to wyświetlany jest komunikat "DOS -> DIR D:*.*" i sterowanie zostaje przekazane do Basica.
5. Jeżeli na dyskietce nie ma pliku DUP.SYS, to sterowanie jest przekazywane do Basica.
6. Jeżeli na dyskietce nie ma pliku MEM.SAV lub jest on zabezpieczony przed zapisem, to do rejestrów DOSVEC, DOSINI i MEMLO zostają przepisane ich wartości oryginalne.
7. Wywołanie procedury odczytu pliku DUP.SYS i przekazanie sterowania do tego pliku.

Jarosław KULIŃSKI

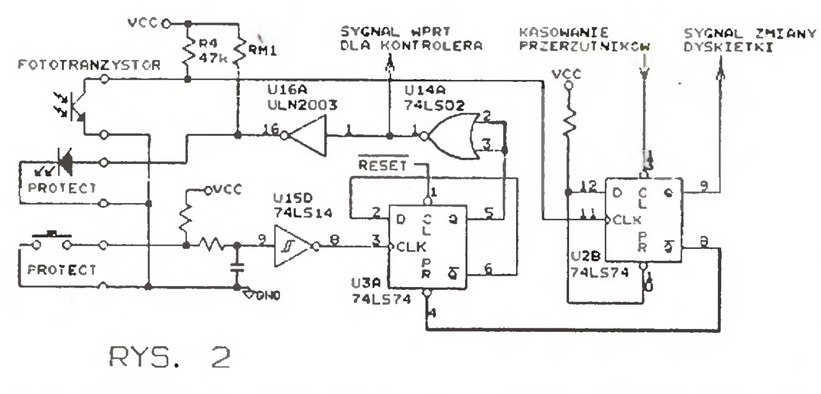
przed przypadkowym zapisem, jak też umożliwi jej zapisanie po drugiej stronie bez wykonywania potrzebnego wycięcia.

Modyfikacja ta polega na zlikwidowaniu dwóch istniejących połączeń i dodaniu dwóch nowych. Po takiej przeróbce każdorazowe włożenie dyskietki do stacji powoduje zaświecenie się wskaźnika WRITE PROTECT, co jest równoznaczne z uniemożliwieniem zapisu, czy też formatowania. Dzieje się tak niezależnie od tego, czy w dyskietce jest wycięcie, czy też nie. Dopiero teraz, również niezależnie od istnienia wycięcia, można naciskając guzik PROTECT skasować zabezpieczenie przed zapisem i tym samym przyciskiem je przywrócić. Dzięki temu użytkownik nie musi pamiętać o zalepieniu istniejącego wycięcia, jak też nie musi robić takowego, aby dokonać zapisu na drugiej stronie dyskietki.

Na rysunku nr 1 przedstawiono fragment schematu ideowego stacji,pokazujący istniejący układ WRITE PROTECT wraz z sygnalizatorem zmiany dyskietki. Kiedy fototranzystor przewodzi, a dzieje się tak, gdy nie ma dyskietki w stacji, to na wejściu 2 bramki U14A jest stan niski, dzięki czemu zmiany na wejściu 3 będą wpływać na stan wyjścia 1 połączonego z kontrolerem stacji. Wejście 3 jest sterowane przez przerzutnik typu D, który przez zwarcie wyjścia NIE-Q z wejściem D pracuje jako J-K. Impulsy zegarowe dla tego przerzutnika powstają w wyniku użycia przycisku



RYS. 1



RYS. 2

PROTECT, dzięki czemu można zabronić zapisu na dyskietce z wycięciem. Jeżeli fototranzystor zostanie zasłonięty przez wsuwaną dyskietkę, to na wejściu 11 przerzutnika U2B stan niski zmieni się na wysoki, co wytworzy informację dla pozostałych układów stacji, że nastąpiło wprowadzenie dyskietki. Aby zmienić działanie opisanego układu, należy zlikwidować połączenia zaznaczone na rys. nr 1 linią przerywaną i według rys. nr 2 wykonać nowe tak, jak zostały one pokazane linią pogrubioną. Teraz fototranzystor ma wpływ tylko na stan przerzutnika U2B, tak, że wsunięcie dysku powoduje jego ustawienie, co pociąga za sobą zaprogramowanie przerzutnika U3A. W wyniku tego na wyjściu bramki U14A, połączonym z wejściem WPRT kontrolera, pojawi się stan niski blokując zapis i formatowanie. Stan przerzutnika U3A można zmienić przez podanie na wejście zegarowe impulsu pochodzącego z bramki U15D sterowanej przyciskiem PROTECT, a więc tym samym zezwolić na zapis lub formatowanie.

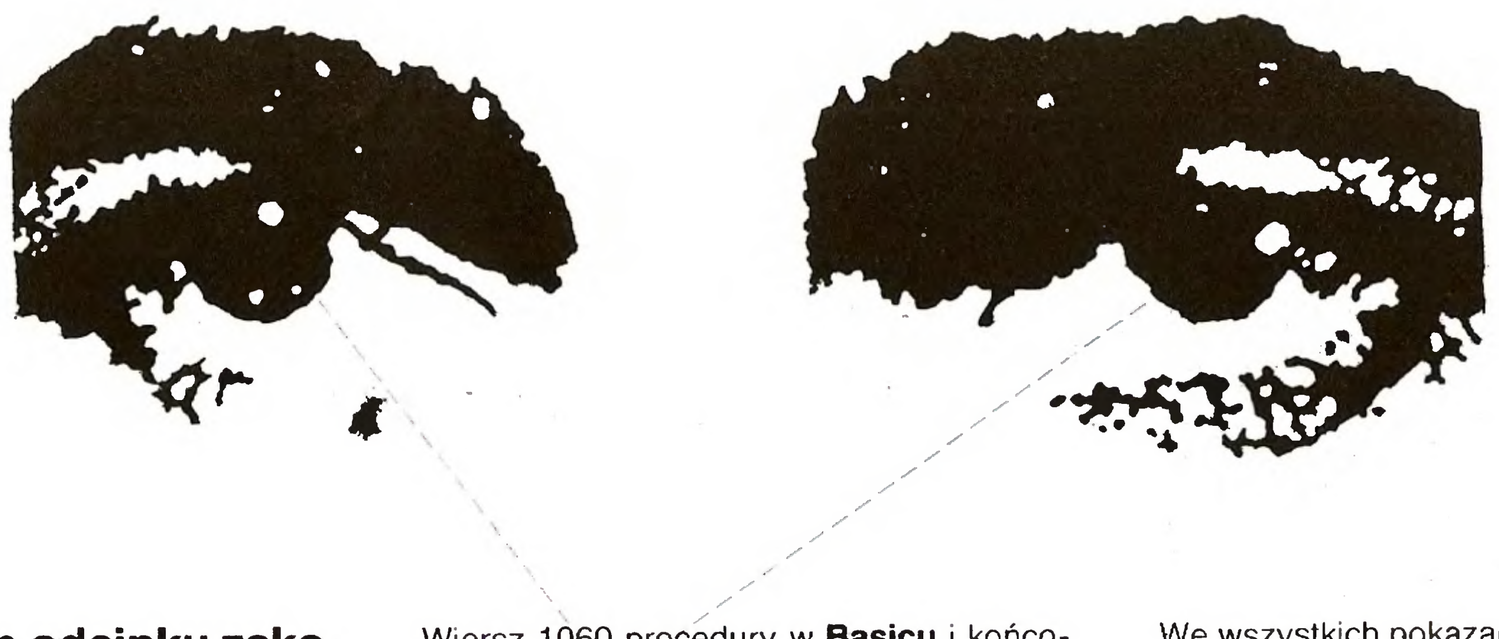
Wykonanie opisywanej przeróbki jest zupełnie proste, trochę problemów może nastręczyć jedynie rozebranie stacji w celu dostania się do płytki drukowanej. Gdy już wyciągniemy płytkę ze stacji, musimy odnaleźć na niej układy oznaczone jako U14, U2 i U3. Nóżkę nr 2 układu U14 należy wylutować z płyty i połączyć z nóżką nr 3 tego samego układu, nóżkę nr 4 układu U3 też trzeba odłączyć od płytki i przy pomocy kawałka cienkiego przewodu połączyć z nogą nr 8 układu U2. Są to wszystkie czynności, jakie musimy wykonać, aby zmienić sposób ochrony dyskietki przed przypadkowym zapisem.

Mariusz GEISLER

SPRZĘT

SZKÓŁKA

JAK KODOWAĆ? (3)



W poprzednim odcinku zakodowaliśmy już podstawowy szkielet gry. Pozostało jednak jeszcze jego całkowite wypełnienie. Dopiero teraz rozpoczyna się najtrudniejszy etap pracy, trzeba tu bowiem rozwiązać wiele problemów.

Pierwszym krokiem musi być określenie wyglądu planszy, na której będziemy rozgrywać naszą zabawę. Zgodnie z założeniami ma to być pole o rozmiarach 10 na 10. Każdy punkt tego pola oznaczymy kropką. Ponadto trzeba na planszy umieścić opisy ułatwiające grę. Będzie to przede wszystkim oznakowanie współrzędnych, a także kierunków. Najprostsza plansza może więc wyglądać tak:

```
          <N>
     0123456789
   0 ..........
   1 ..........
   2 ..........
   3 ..........
   4 ..........
<W>5 ..........  <E>
   6 ..........
   7 ..........
   8 ..........
   9 ..........
          <S>
```

Przełóżmy to teraz na oba języki programowania, w których kodujemy nasz program. Początek jest oczywisty: należy wybrać tryb graficzny i określić kolory. Ponieważ gra jest bardzo prosta, to wystarczy tryb 0, a ponieważ jest morska, to tło zrobimy niebieskie. Dodatkowo wyłączymy kursor, aby nie przeszkadzał nam na ekranie i zmienimy margines, aby plansza znalazła się na środku ekranu. Po tym wstępie można już narysować planszę wraz z opisem. W **Action!** efekt będzie następujący:

```
PROC Plansza()
 BYTE lmarg=82,i
  Graphics(0)
  SetColor(2,8,0)
  crsinh=1
  lmarg=14
  Put(125) PutE() PutE()
  PrintE("  0123456789")
  PutE()
  FOR i=0 TO 9 DO
    PrintB(i) PrintE(" ..........")
  OD
  lmarg=2
  Position(19,0) PrintE("<N>")
  Position(19,15) PrintE("<S>")
  Position(8,8) PrintE("<W>")
  Position(28,8) PrintE("<E>")
  Okret(mx,my)
  FOR i=1 TO 2 DO
    Niszczyciel(i)
  OD
  Statek(sx,sy)
  Czyszczenie()
RETURN
```

W **Basicu** zaś taki:

```
SH    997 REM
ZB    998 REM *** Plansza ***
SN    999 REM
KE    1000 GRAPHICS 0:SETCOLOR 2,8,0:POKE 75
      2,1:POKE 82,14:? CHR$(125):? :? "  012
      3456789":?
PQ    1010 FOR I=0 TO 9:? I;" ..........":NE
      XT I:POKE 82,2
GV    1020 POSITION 19,0:? "<N>"
PJ    1030 POSITION 19,15:? "<S>"
VM    1040 POSITION 8,8:? "<W>"
EJ    1050 POSITION 28,8:? "<E>"
DP    1060 GOSUB 60:FOR N=0 TO 1:GOSUB 70:NE
      XT N:GOSUB 80:GOSUB 40:RETURN
```

Zwróć uwagę na konieczność odtworzenia wartości marginesu przed opisywaniem kierunków — spróbuj zamienić te operacje miejscami i zobacz, co się wtedy stanie.

Wiersz 1060 procedury w **Basicu** i końcowy fragment procedury w **Action!** zawierają dodatkowe elementy, o których wcześniej nic nie napisałem. Są to wywołania procedur umieszczających na gotowej planszy jednostki biorące udział w grze. Ich początkowe położenie zostało już wcześniej określone podczas inicjowania.

Dla okrętu podwodnego wybierzemy symbol „+" (kod ASCII = 43). Współrzędne określające jego położenie przekażemy jako parametry do następującej procedury:

```
PROC Okret(BYTE x,y)
  GetCur()
  Position(x+16,y+4)
  Put(43)
  PutCur()
RETURN
```

W **Basicu** zamiast przekazywania parametrów trzeba wykorzystać odpowiednie zmienne globalne, więc procedura będzie wyglądała tak:

```
BS    58 REM
VD    59 REM *** Okret podwodny ***
NX    60 GOSUB 20:POSITION MX+16,MY+4:? "+";
      :GOSUB 30:RETURN
```

W analogiczny sposób umieścimy na planszy statek oznaczony symbolem „#" (kod ASCII = 35). W **Action!:**

```
PROC Statek(BYTE x,y)
  GetCur()
  Position(x+16,y+4)
  Put(35)
  PutCur()
RETURN
```

i w **Basicu:**

```
BU    78 REM
OH    79 REM *** Statek ***
OL    80 GOSUB 20:POSITION SX+16,SY+4:? "#";
      :GOSUB 30:RETURN
```

Niszczyciel będzie symbolizowany znakiem „*" (kod ASCII = 42). Ponieważ są dwie takie jednostki, to parametrem procedury w **Action!** musi być numer niszczyciela, współrzędne zostaną zaś pobrane z tablicy globalnej:

```
PROC Niszczyciel(BYTE n)
  GetCur()
  Position(dx(n)+16,dy(n)+4)
  Put(42)
  PutCur()
RETURN
```

W **Basicu** będzie to niemal identycznie (poza przekazaniem parametru):

```
BT    68 REM
AK    69 REM *** Niszczyciel ***
UU    70 GOSUB 20:POSITION DX(N)+16,DY(N)+4:
      ? "*";:GOSUB 30:RETURN
```

Dzięki takiej konstrukcji procedury rysującej niszczyciel można ją wykorzystać dla dowolnej z jednostek tego typu. W procedurze rysującej planszę umieszczenie obu zostało zrealizowane w pętli FOR.

Podczas gry wszystkie jednostki przemieszczają się, więc potrzebna będzie z pewnością procedura, która w miejscu symbolu okrętu narysuje ponownie kropkę (kod ASCII = 46). Napiszmy ją od razu, choć nie występuje podczas rysowania planszy. W **Action!** będzie to tak:

```
PROC Kasowanie(BYTE x,y)
  GetCur()
  Position(x+16,y+4)
  Put(46)
  PutCur()
RETURN
```

a tak w **Basicu:**

```
BR    48 REM
NG    49 REM *** Kasowanie napisu ***
VW    50 GOSUB 20:POSITION OX+16,OY+4:? ".";
      :GOSUB 30:RETURN
```

We wszystkich pokazanych wyżej procedurach współrzędne jednostek są zwiększane o 16 (poziomo) lub 4 (pionowo). Jest to konieczne, aby symbol jednostki znalazł się we właściwym miejscu planszy. W tym celu zwiększenie współrzędnych musi być równe przesunięciu planszy od lewego górnego narożnika ekranu.

Nie omówionymi elementami powyższych procedur są również wywołania na ich początku i końcu. Ponieważ podczas gry jednostki będą rysowane na planszy w trakcie wyświetlania komunikatów pod planszą, to niezbędne jest zapamiętywanie położenia kursora. W przeciwnym razie komunikaty byłyby wyświetlane częściowo na planszy. Pierwsza procedura zapamiętuje więc aktualne położenie kursora w zmiennych TX i TY, co w **Basicu** jest zrealizowane następująco:

```
BO    18 REM
BD    19 REM *** Get Cursor ***
CE    20 TX=PEEK(85):TY=PEEK(84):RETURN
```

w **Action!** zaś (zmienne TX i TY muszą być globalne, gdyż nie można zwrócić z procedury dwóch wartości):

```
PROC GetCur()
  tx=crscol
  ty=crsrow
RETURN
```

Następnie symbol jednostki jest rysowany w odpowiednim miejscu, po czym kolejna procedura umieszcza kursor na poprzedniej pozycji. W **Basicu** tak:

```
BP    28 REM
NI    29 REM *** Put Cursor ***
FA    30 POKE 85,TX:POKE 84,TY:RETURN
```

a w **Action!** tak:

```
PROC PutCur()
  crscol=tx
  crsrow=ty
RETURN
```

Na koniec pozostała jeszcze procedura usuwająca z ekranu poniżej planszy wszystkie dotychczas wyświetlone komunikaty. Najprościej będzie zrealizować taką operację przez umieszczenie kursora w pierwszym wierszu komunikatów i kilkakrotne wpisanie znaku kasowania wiersza (kod ASCII = 156). W tym przypadku wygląd procedury w **Action!:**

```
PROC Czyszczenie()
 BYTE i
  Position(2,17)
  FOR i=0 TO 7 DO
    Put(156)
  OD
RETURN
```

i w **Basicu:**

```
BQ    38 REM
HZ    39 REM *** Czyszczenie ekranu ***
YO    40 POSITION 2,17:FOR I=0 TO 7:? CHR$(1
      56);:NEXT I:RETURN
```

jest identyczny (pomijając sposób zapisu instrukcji).

W ten sposób mamy już przygotowane wszystko, co potrzeba do rozpoczęcia gry, a ściślej mówiąc do rozpoczęcia jej zapisywania w postaci programu. To jednak będzie tematem następnych odcinków „Szkółki".

Wojciech ZIENTARA

KRÓTKIE, KRÓTSZE, NAJKRÓTSZE...

Wiele spośród listów, które nadchodzą do naszej redakcji, zawiera krótkie programy pisane przez Czytelników. Trudno je publikować w postaci oddzielnych artykułów, a szkoda by było odsyłać do redaktora Kosha. Otwieramy więc nową rubrykę, w której będziemy drukować takie właśnie perełki otrzymane od naszych Czytelników.

Jako pierwszy zamieszczamy programik nadesłany z Czechosłowacji przez Romana Plaka. Służy on do zamiany liczb arabskich na rzymskie. Przeznaczoną do zamiany liczbę arabską należy wpisać na wyświetlone żądanie, a niemal natychmiast ukazuje się jej odpowiednik w postaci liczby rzymskiej.

```
QM    1 REM ZAMIANA ARABSKIE -> RZYMSKIE
RF    2 REM Roman Plak
DV    3 REM (c) 1991, Sp. Bajtek
UP    10 DIM X$(7),P1$(1),P2$(1),P3$(1),A(4)
      :X$="MDCLXVI"
BA    20 INPUT C:FOR I=1 TO 3:READ H:A(I)=IN
      T(C/H):C=C-A(I)*H:NEXT I:A(4)=C
OH    30 FOR I=1 TO 4:IF A(I)=0 THEN 100
NP    40 P1$=X$(I*2-1,I*2-1):IF I=1 THEN GOS
      UB 110:GOTO 100
KR    50 P2$=X$(I*2-2,I*2-2):P3$=X$(I*2-3,I*
      2-3):IF A(I)<4 THEN GOSUB 110
NM    60 IF A(I)=4 THEN ? P1$;P2$;
UZ    70 IF A(I)=5 THEN ? P2$;
BS    80 IF A(I)>5 AND A(I)<9 THEN ? P2$;:A(
      I)=A(I)-5:GOSUB 110
QX    90 IF A(I)=9 THEN ? P1$;P3$;
EU    100 NEXT I:? :END
QX    110 FOR J=1 TO A(I):? P1$;:NEXT J:RETU
      RN
JJ    120 DATA 1000,100,10
QK    130 REM ****
```

Drugi programik, napisany przez Władysława Zwierzchowskiego z Lublina (SP8DX0), jest przeznaczony dla amatorów krótkofalarstwa. Umożliwia on wizualne kontrolowanie jakości nadawania alfabetem Morse'a. Klucz telegraficzny należy przyłączyć do styków 1 i 8 w pierwszym gnieździe joysticka.

```
AB    1 REM KONTROLA KLUCZOWANIA
IL    2 REM Wladyslaw Zwierzchowski SP8DXO
DV    3 REM (c) 1991, Sp. Bajtek
RL    10 GRAPHICS 8:COLOR 15
ZB    20 FOR A=0 TO 155 STEP 5
QQ    30 FOR B=0 TO 319
EC    40 SOUND 0,150,10,150-STICK(0)*10
RN    50 COLOR 15-STICK(0)
YJ    60 PLOT B,A
HG    70 NEXT B:NEXT A
TF    80 REM ****
```

Na temat blokowania działania klawisza ‹RESET› powiedziano już bardzo dużo. Ale jeszcze nie wszystko, co udowadnia program nadesłany przez Radosława Kuberę z Koszalina. Każdorazowo, po naciśnięciu ‹RESET› powoduje on umieszczenie w rejestrze CHBASE adresu zestawu znaków zdefiniowanego przez użytkownika (szósta liczba w instrukcji DATA) oraz ustawia kolory ekranu według wartości zapisanych w komórkach 1536—1538 (trzy pierwsze liczby w DATA).

```
LU    1 REM BLOKADA RESET
DU    2 REM Radoslaw Kubera
DV    3 REM (c) 1991, Sp. Bajtek
ZH    10 FOR I=1536 TO 1563:READ A:POKE I,A:
      NEXT I
CD    20 POKE 9,1:POKE 12,4:POKE 13,6
BQ    30 DATA 15,0,0,104,169,152,141,244,2,1
      73,0,6,141,197,2,173,1,6,141,198,2,173
      ,2,6,141,200,2,96
TB    40 REM ****
```

Interesujący programik otrzymaliśmy od Tomasza Bielaka. Po jego przepisaniu i uruchomieniu zobaczysz wiersz oznaczony numerem 1000. Uruchom go i naciśnij dowolny klawisz. Zmienna KEY otrzyma wartość odpowiadającą kodowi ATASCII naciśniętego klawisza. Oznacza to, że instrukcja zawarta w wierszu 1000 odpowiada sekwencji instrukcji **OPEN #1,4,0,„K:":GET #1,KEY:CLOSE #1** nie wykorzystując przy tym żadnego kanału IOCB.

```
CL    1 REM GET KEY
XR    2 REM Tomasz Bielak
DV    3 REM (c) 1991, Sp. Bajtek
LP    10 GRAPHICS 0
FW    20 POSITION 2,8:? "1000 KEY=USR(ADR(":
      CHR$(34);
BM    30 FOR D=1 TO 11:READ A:? CHR$(27);CHR
      $(A);:NEXT D:? CHR$(34);"))"
CZ    40 POSITION 2,4:? "NEW"
RT    50 POSITION 2,12:? "POKE 842,12:? CHR$
      (125):L."
IG    60 POKE 842,13:POSITION 0,0:STOP
IV    70 DATA 104,32,54,254,133,212,169,0,13
      3,213,96
TF    80 REM ****
```

Często przy przeglądaniu dłuższych wydruków posługujemy się linijką dla oddzielenia wierszy. Rozwiązanie to nie daje się jednak zastosować na ekranie monitora, chyba że będzie to linijka wykonana według programu, którego autorem jest Marek Omirski. Do jej przesuwania służy klawisz ‹HELP›. Procedura wykorzystuje przerwania zegara systemowego i nie jest relokowalna.

```
JB    1 REM LINIJKA
ON    2 REM Marek Omirski
DV    3 REM (c) 1991, Sp. Bajtek
OT    10 FOR I=1536 TO 1611:READ A:POKE I,A:
      S=S+A:NEXT I
LW    20 IF S<>7139 THEN ? "BLAD W DANYCH!":
      END
TG    30 I=USR(1536)
JD    40 DATA 104,169,22,141,40,2,169,6,141,
      41,2,169,8,141,26,2,169,64,141,14,212,
      96,173,220,2,240,43,238,33,6
YE    50 DATA 169,32,141,38,156,238,41,6,169
      ,2,141,37,156,173,33,6,201,60,208,15,1
      69,38,141,33,6,169,2,141,60
LD    60 DATA 156,169,37,141,41,6,169,0,141,
      220,2,169,8,141,26,2,96
TE    70 REM ****
```

Zapraszamy wszystkich Czytelników do przesyłania nam takich króciutkich programów. Autor każdego opublikowanego programu otrzyma honorarium w wysokości 50000 zł. Jednocześnie prosimy o przesyłanie tekstów źródłowych procedur w języku maszynowym (jak użyte w trzech powyższych programikach).

Wojciech ZIENTARA

DUSZKI W NATARCIU

LISTING 1

```
WN    0 REM ********************************
ZA    1 REM *                              *
OT    2 REM * PROGRAM NR.1 - GENERATOR    *
KF    3 REM * PROCEDUR MASZYNOWYCH PMG     *
ZD    4 REM *                              *
WW    5 REM * Jakub Cebula                 *
VK    6 REM * (c) 1990, Sp. Bajtek         *
ZG    7 REM *                              *
WV    8 REM ********************************
WM   10 DIM PLMOV$(66),PMI$(103)
AK   20 GRAPHICS 0:POKE 559,0:POSITION 2,2
QC   30 ? "NEW":?
SE   40 LIST 10
GV   50 ? :? "20 PMI$=";CHR$(34);
XX   60 FOR I=1 TO 103:READ A:? CHR$(27);CH
     R$(A);:NEXT I:? CHR$(34)
KZ   70 ? :? "30 PLMOV$=";CHR$(34);
ZZ   80 FOR I=1 TO 66:READ A:? CHR$(27);CHR
     $(A);:NEXT I:? CHR$(34)
XH   90 ? :? "GR.0:POKE 842,12":POSITION 0,
     0:POKE 842,13:END
KT  100 REM ------------------------------
DY  110 DATA 104,173,230,2,133,206,56,233,
     8,72,165,106,56,237,230,2,201,8,240,7,
     144,5,56
CK  120 DATA 233,8,208,245,133,204,169,8,5
     6,229,204,133,204,104,56,229,204,133,2
     04,133,207,169,0
KI  130 DATA 133,203,133,205,160,0,169,0,1
     45,203,24,165,203,105,1,133,203,165,20
     4,105,0,133,204,197,206
SP  140 DATA 144,235,169,3,141,29,208,165,
     207,141,7,212,133,204,230,204,230,204,
     230
ML  150 DATA 204,230,204,169,0,133,203,169
     ,58,141,47,2,96
RI  160 DATA 104,165,204,133,207,104,104,1
     70,101,204,133,204,160,0,169,0,145,203
     ,200,208,251,104,104,157,0
IR  170 DATA 208,104,104,133,206,104,133,2
     06,104,133,205,160,0,177,205,133,209,2
     00,177
LU  180 DATA 205,72,152,170,104,164,208,14
     5,203,230,208,138,168,196,209,144,237,
     165,207,133,204,96
```

LISTING 2

```
EZ   0 REM ----------------------------------
MQ   1 REM PROGRAM NR.2
JY   2 REM WYNIK DZIALANIA PROGRAMU NR.1
QY   3 REM Jakub Cebula
DI   4 REM (c) 1990, Sp. Bajtek
FE   5 REM ----------------------------------
YZ  10 DIM PMI$(103),PLMOV$(66)
OL  20 PMI$="h■f ┌N8i┘H%j8Mf I┘p\&┐8i┘PU┐L
     )┘8eL┐Lh8eL┐L┐O)♥┐K┐M ♥)♥┌K┼%Ki ├┐K%Li♥
     ┌LEN&k)┘ ■↓P%O ┐\T┐L┌fL┌fL┌fL┌f)♥┐K):■/ |♦"
IM  30 PLMOV$="h%L┐Ohh%eL┐L ♥)♥┌KHP♣hh↓♥Ph
     h┐Ph┐Nh┐M ♥1M┐QH1MH┐%hSP┌KfP ┐()8&m%O┐L
     ♦"
```

LISTING 3

```
CZ   0 ----------------------------------
ZB   1 REM PROGRAM DEMONSTRACYJNY
QX   2 REM Jakub Cebula
DH   3 REM (c) 1990, Sp. Bajtek
NJ   4 REM
IO   5 REM Linie 10,30 powstaly w wyniku
JA   6 REM dzialania programu nr.1
NM   7 REM
DH   8 ----------------------------------
FI  10 DIM PMI$(103),PLMOV$(66),MI$(22),FW
     $(22)
OL  20 PMI$="h■f ┌N8i┘H%j8Mf I┘p\&┐8i┘PU┐L
     )┘8eL┐Lh8eL┐L┐O)♥┐K┐M ♥)♥┌K┼%Ki ├┐K%Li♥
     ┌LEN&k)┘ ■↓P%O ┐\T┐L┌fL┌fL┌fL┌f)♥┐K):■/ |♦"
IM  30 PLMOV$="h%L┐Ohh%eL┐L ♥)♥┌KHP♣hh↓♥Ph
     h┐Ph┐Nh┐M ♥1M┐QH1MH┐%hSP┌KfP ┐()8&m%O┐L
     ♦"
HZ  40 ----------------------------------
LX  50 REM DANE DO ZMIENNYCH
NQ  60 FOR I=1 TO 22:READ A:MI$(I,I)=CHR$(
     A):NEXT I
UY  70 FOR I=1 TO 22:READ A:FW$(I,I)=CHR$(
     A):NEXT I
AJ  80 GRAPHICS 0:X=USR(ADR(PMI$)):REM STA
     RT GRAFIKI PMG
UA  90 POKE 704,14:REM KOLOR GRACZA 0
QE 100 FW=ADR(FW$):MI=ADR(MI$):REM TABLIC
     E DANYCH
OQ 110 ----------------------------------
VN 120 DO
EU 130 FOR I=0 TO 255
OF 140 IF K=1
FY 150 K=0:X=USR(ADR(PLMOV$),0,I,60,FW)
MW 160 X=USR(ADR(PLMOV$),1,I,120,MI)
MZ 170 ELSE
WT 180 K=1:X=USR(ADR(PLMOV$),0,I,60,MI)
VZ 190 X=USR(ADR(PLMOV$),1,I,120,FW)
IJ 200 ENDIF
WT 210 PAUSE 3
FW 220 NEXT I
RB 230 LOOP
OX 240 ----------------------------------
ID 250 REM DUSZEK MARSZ
LK 260 DATA 21,16,56,124,254,56,52,56,16,
     56,116,112,250,255,125,124,124,124,198
     ,198,198,231
IL 270 REM DUSZEK STOP
GQ 280 DATA 21,16,56,124,254,56,52,56,16,
     56,124,108,108,108,108,108,124,56,24,2
     4,28,62
```

Wydawać by się mogło, że o P/MG napisano już wszystko. Drogi Czytelniku, to złudzenia! Dziś trochę o ruchu i animacji.

Osobiście jestem zwolennikiem przerzucania jak największej części pracy na komputer, dlatego przedstawiam procedury maszynowe, które za Ciebie uruchomią grafikę P/M i zapewnią Ci nad nią kontrolę. Przy odrobinie wysiłku będziesz mógł animować zaprojektowane postacie we wszystkich trybach graficznych.

Pierwsza z procedur zajmuje się uruchomieniem duszków. Po wywołaniu, sama wybiera obszar na grafikę P/M i kasuje go. Ustawia ona również wszystkie parametry niezbędne do pracy z duszkami. Należy ją wywoływać każdorazowo po instrukcji **GRAPHICS**. Obszar zajęty przez grafikę P/M nie jest chroniony! Oznacza to, że w rzeczywistości mamy od 2 do 4 KB (zależy to od trybu graficznego) mniej pamięci niż podaje funkcja FRE — obszar ten znajduje się bezpośrednio poniżej pamięci obrazu. Rozwiązanie to ma dwie poważne zalety: nie zmienia położenia pamięci obrazu i znacznie skraca procedurę

uruchamiania grafiki. Wywołanie funkcji ma postać **X=USR(ADR(PMI$))**.

Druga procedura wyświetla duszka na ekranie. Pierwszy bajt wzoru gracza określa jego długość. Wywołanie ma postać **X=USR(ADR(PLMOV$), nr duszka (0-3), pozycja pozioma, pozycja pionowa, adres tablicy z danymi duszka).** Przypominam, że kolor obiektu określa się instrukcją **POKE 704+n,col**, gdzie n to numer duszka, a col — kod koloru. Teraz przesuwanie duszka jest czynnością bardzo prostą, a równie łatwa jest animacja. Prezentuje to program demonstracyjny (listing 3), napisany w Turbo Basicu. Narzucającym się, wygodnym rozwiązaniem jest umieszczenie duszka w zmiennych tekstowych, co też uczyniono. Ze względu na możliwość wydruku dane te są odczytywane z instrukcji **DATA**, czego oczywiście w swoich programach czytelnik robić nie musi. Proszę zauważyć, że po ekranie spacerują dwie postacie, choć zdefiniowano tylko jedną — to tylko cząstka z możliwości, jakie dają te procedury.

Aby uzyskać procedury maszynowe z listingu 2, należy wpisać listing 1 (na wszelki wypadek go zapisać) i wykonać **RUN**. Wynikiem będzie program, który można zapisać instrukcją **LIST** i dołączać do swoich programów. Oprócz tego dla ciekawskich podaję treść procedur w asemblerze MAC/65. Procedura inicjowania jest pokazana na listingu 4, a listing 5 przedstawia procedurę ruchu i animacji.

Jakub CEBULA

LISTING 4

```
0100 ;P/MG Init
0110 ;Jakub Cebula
0120 ;(c) 1990, Sp. Bajtek
0130 ;
0140      *=  $0600
0150 ;
0160 RAMTOP = 106
0170 SDMCTL = 559
0180 MEMTOP = 741
0190 PMCNTL = 53277
0200 PMBASE = 54279
0210 ;
0220      PLA
0230      LDA MEMTOP+1
0240      STA 206
0250      SEC
0260      SBC #8
0270      PHA
0280      LDA RAMTOP
0290      SEC
0300      SBC MEMTOP+1
0310 X1   CMP #8
0320      BEQ EX
0330      BCC EX
0340      SEC
0350      SBC #8
0360      BNE X1
0370 EX   STA 204
0380      LDA #8
0390      SEC
0400      SBC 204
0410      STA 204
0420      PLA
0430      SEC
0440      SBC 204
0450      STA 204
0460      STA 207
0470      LDA #0
0480      STA 203
0490      STA 205
0500      LDY #0
0510 X    LDA #0
0520      STA (203),Y
0530      CLC
0540      LDA 203
0550      ADC #1
0560      STA 203
0570      LDA 204
0580      ADC #0
0590      STA 204
0600      CMP 206
0610      BCC X
0620      LDA #3
0630      STA PMCNTL
0640      LDA 207
0650      STA PMBASE
0660      STA 204
0670      INC 204
0680      INC 204
0690      INC 204
0700      INC 204
0710      LDA #0
0720      STA 203
0730      LDA #58
0740      STA SDMCTL
0750      RTS
```

LISTING 5

```
0100 ;Player Movement
0110 ;Jakub Cebula
0120 ;(c) 1990, Sp. Bajtek
0130 ;
0140      *=  1700
0150 ;
0160 HPOSPO = 53248
0170      PLA
0180      LDA 204
0190      STA 207
0200      PLA
0210      PLA
0220      TAX
0230      ADC 204
0240      STA 204
0250      LDY #0
0260      LDA #0
0270 X    STA (203),Y
0280      INY
0290      BNE X
0300      PLA
0310      PLA
0320      STA HPOSPO,X
0330      PLA
0340      PLA
0350      STA 208
0360      PLA
0370      STA 206
0380      PLA
0390      STA 205
0400      LDY #0
0410      LDA (205),Y
0420      STA 209
0430 X1   INY
0440      LDA (205),Y
0450      PHA
0460      TYA
0470      TAX
0480      PLA
0490      LDY 208
0500      STA (203),Y
0510      INC 208
0520      TXA
0530      TAY
0540      CPY 209
0550      BCC X1
0560      LDA 207
0570      STA 204
0580      RTS
```

Magnetofon jako pamięć zewnętrzna komputera należy niewątpliwie do muzeum informatyki. Mimo to jest on w dalszym ciągu bardzo popularny w naszym kraju ze względu na niską — w stosunku do innych urządzeń — cenę.

Niestety jest to jedyna jego zaleta. Standardowy system komunikacji Atari z magnetofonem jest po prostu fatalny. Prędkość transmisji 600 bodów, 128-bajtowe rekordy, długi sygnał pilotujący — oto główne wady zapisu na kasecie.

Na domiar złego przy pomocy środków programowych niewiele można zdziałać. Proponuję jednak uważnie przyjrzeć się systemowym procedurom, gdyż niejednego można się z nich nauczyć (są one szczegółowo opisane w książce Wojciecha Zientary „Mapa pamięci Atari. Procedury wejścia/wyjścia").

Procedury obsługi magnetofonu zajmują w pamięci ROM obszar od adresu $FCDB do $FE8C. Oto one:

CASOPN ($FCE6=64742) — przygotowuje magnetofon do zapisu lub odczytu, tj. uruchamia silnik, zapisuje sygnał pilotujący itd.;

CASRDBT ($FD7A=64890) — odczytuje jeden bajt z bufora magnetofonu, a po jego wyczerpaniu wywołuje CASRDBL;

CASRDBL ($FD8D=64909) — odczytuje blok (rekord) z kasety i umieszcza go w buforze magnetofonu (CASBUF $03FD—$047F);

CASWRT ($FDB4=64948) — przesyła bajt do bufora magnetofonu i po jego zapełnieniu zapisuje rekord na taśmie;

CASCLS ($FDCF=64975) — przy zapisie wysyła pozostałe w buforze dane, dopisuje znak końca pliku (EOF) i wyłącza silnik magnetofonu, przy odczycie tylko wyłącza silnik;

BPWT ($FDFC=65020) — wywołuje pojedynczy (przy odczycie) lub podwójny (przy zapisie) dźwięk brzęczyka i oczekuje na naciśnięcie klawisza;

SYSBUF ($FE3F=65087) — najważniejsza procedura obsługi magnetofonu — tu realizowany jest zapis i odczyt;

WSIOSB ($FE7C=65148) — zapisuje rekord o typie określonym przez zawartość akumulatora.

Oprócz bloków IOCB służących do komunikacji z urządzeniami zewnętrznymi system operacyjny posiada jeszcze tzw. Device Control Block. Jego budowa podobna jest do normalnego IOCB i przeznaczenie zbliżone — obsługa IOCB dla magnetofonu, stacji dysków, drukarki i interfejsu RS232 sprowadza się głównie do wykorzystania DCB. Rozpoczyna się on od adresu $0300 i zawiera:

DDEVIC ($0300) — tu umieszcza się kod rozpoznawczy magnetofonu ($60), stacji dysków ($30), drukarki ($40) lub RS232 ($50);

DUNIT ($0301) — numer wzywanego urządzenia;

DCMND ($0302) — kod przeprowadzanej operacji: $52 — odczyt, $50 — zapis sektora bez weryfikacji, $57 zapis rekordu (na kasecie) lub zapis sektora z weryfikacją (na dyskietce);

DSTATS ($0303) — status operacji, procedura SYSBUF umieszcza tu wartość oznaczającą zapis ($80) lub odczyt ($40);

DBUFA ($0304—$0305) — adres początkowy bufora danych;

DTIMLO ($0306) — liczba sekund do błędu TIMEOUT;

DBYT ($0308—$0309) — długość bufora ustalana przez SYSBUF na 131 bajtów;

DAUX1 i 2 ($030A, $030B) — przy komunikacji ze stacją dysków podaje się tu numer sektora, a dla magnetofonu DAUX2 określa długość przerw między rekordami (0 — długie, 128 — krótkie).

Jeżeli chcemy zapisać sektor na dyskietce, to po ustaleniu odpowiednich parametrów należy wywołać procedurę DSKINT ($E453). Natomiast dla zapisania rekordu o długości podanej w DBYT wykonujemy skok do SIOINT ($E459). W ten sposób można nagrywać dane bez przerw, zmieniać długość rekordu itp. Tu jedna uwaga — pierwsze dwa bajty zapisywanego bloku muszą być równe $55, bo inaczej nie uda się nam później go odczytać. Bajty te są wykorzystywane przy synchronizacji transmisji i pomiarze prędkości odczytu, ponadto suma kontrolna jest inicjowana wartością $AA ($55+$55).

Komentarz do wymienionych procedur jest raczej lakoniczny, gdyż są one na tyle proste, że bez problemu zrozumieją je wszyscy zainteresowani. Zwracam w tym miejscu szczególną uwagę na SYSBUF oraz główną wadę wszystkich procedur — jednobajtowy licznik BPTR ($3D) — według niego zapisywane są dane w buforze, ich liczba jest więc ograniczona do 255 w jednym rekordzie.

Spróbujmy wykorzystać zdobyte wiadomości. Jedyny prezentowany program (listing 2 jest jego tekstem źródłowym) pozwala zdefiniować własny format zapisu: ustalić długość rekordu (do 255) i prędkość zapisu (do 900 bodów) oraz skrócić sygnał pilotujący do 5 sekund, co daje znaczne oszczędności. Zapisuje on na kasecie krótki plik typu BOOT, który przepisuje ROM do RAM i wprowadza odpowiednie zmiany w systemie operacyjnym. Jeżeli chcemy powrócić do standardowego formatu, wpisujemy POKE 54017,253. Po naciśnięciu ‹RESET› nowy, własny format przywracamy przez POKE 54017,254.

W następnym odcinku zajmiemy się zabezpieczaniem programów.

Andrzej ZALEWSKI

LISTING 1

```
WG    1 REM TYLKO DLA ZAAWANSOWANYCH (3)
LU    2 REM Andrzej Zalewski
DV    3 REM (c) 1991, Sp. Bajtek
TX    10 GRAPHICS 0:L=990:FOR N=20000 TO 201
      70 STEP 21
VD    20 L=L+10:S=0
HO    30 FOR M=0 TO 20
PN    40 READ A:S=S+A:POKE N+M,A
NY    50 NEXT M:READ SS:IF SS<>S THEN ? "BLA
      D W WIERSZU ";L:END
KN    60 NEXT N
PQ    70 ? "Podaj dlugosc rekordu /do 255/";
      :INPUT N:IF N>255 OR N<3 THEN ? CHR$(2
      53);"[illegible]":GOTO 70
QH    80 NQ=N-1:NN=N+3:N1=INT(NN/256):NO=NN-
      N1*256
AN    90 POKE 20058,N:POKE 20063,NQ
VC    100 POKE 20074,NO:POKE 20079,N1
OR    120 ? "Podaj predkosc zapisu /600-900/
      ";:INPUT N
HY    130 IF N>900 OR N<600 THEN ? CHR$(253)
      ;"[illegible]":GOTO 120
DV    140 NQ=INT(895000/N):N1=INT(NQ/256):NO
      =NQ-N1*256
TU    150 POKE 20089,N1:POKE 20084,NO
JU    180 OPEN #1,8,128,"C:":POKE 850,11
DY    190 POKE 852,32:POKE 853,78:POKE 856,2
      00:POKE 857,0
QL    200 X=USR(ADR("h[illegible]LV[illegible]")):CLOSE #1
CQ    1000 DATA 0,2,250,79,155,80,160,159,16
      9,0,141,68,2,141,144,80,141,23,80,200,
      140,2214
RK    1010 DATA 145,80,140,24,80,162,0,189,0
      ,160,157,0,96,232,208,247,32,116,80,19
      2,207,2547
YK    1020 DATA 208,2,160,215,192,255,240,2,
      208,223,169,254,141,1,211,169,255,141,
      154,253,169,3622
RA    1030 DATA 254,141,167,253,141,190,253,
      141,225,253,169,2,141,72,254,169,1,141
      ,67,254,169,3457
VZ    1040 DATA 160,141,163,235,169,161,141,
      168,235,162,0,189,157,80,157,87,253,15
      7,10,253,232,3310
OC    1050 DATA 224,9,208,242,169,50,141,13,
      253,169,60,141,2,211,76,0,160,120,169,
      254,141,2812
JW    1080 DATA 1,211,169,156,141,250,255,14
      1,254,255,169,80,141,251,255,141,255,2
      55,162,0,189,3731
ST    1070 DATA 0,96,157,0,160,232,208,247,8
      8,169,253,141,1,211,96,64,162,0,160,25
      0,234,2929
UN    1080 DATA 234,234,234,234,234,0,0,0,0,
      0,0,0,0,0,0,0,0,0,0,0,0,1170
```

LISTING 2

```
0100 ;Program poprawy zapisu
0110 ;
0120 COLDST = $0244
0130 TRAM =  $6000
0140 ROM =   $A000
0150 PORTB = $D301
0160 NMIVEC = $FFFA
0170 IRQVEC = $FFFE
0180 ;
0190      *=  $4FFA
0200 ;
0210      .BYTE 0,2,$FA,$4F, <RTS, >RTS
0220 ;
0230 ;przygotowanie
0240      LDY #$9F
0250      LDA #0
0260      STA COLDST
0270      STA GDZIE+1
0280      STA PO+1
0290      ;
0300 ;
0310 ;przenosimy ROM do RAM - $6000
0320 ;RAM do RAM pod ROM
0330 TRS
0340      INY
0350      STY GDZIE+2
0360      STY PO+2
0370      LDX #0
0380 PO   LDA ROM,X
0390      STA TRAM,X
0400      INX
0410      BNE PO
0420      JSR TRBYT
0430      CPY #$CF
0440      BNE CNT
0450      LDY #$D7
0460 CNT  CPY #$FF
0470      BEQ EXIT
0480      BNE TRS
0490 EXIT LDA #$FE
0500      STA PORTB
0510 ;
0520      LDA #$FF
0530      STA 64922
0540      LDA #$FE
0550      STA 64935
0560      STA 64958
0570      STA 64993
0580      LDA #$02
0590      STA 65096
0600      LDA #$01
0610      STA 65091
0620 ;
0630      LDA #$A0
0640      STA 60323
0650      LDA #$A1
0660      STA 60328
0670 ;
0680      LDX #0
0690 AA   LDA DATA,X
0700      STA 64855,X
0710      STA 64778,X
0720      INX
0730      CPX #9
0740      BNE AA
0750      LDA #50
0760      STA 64781
0770      LDA #60
0780      STA 54018
0790      JMP ROM
0800 ;
0810 TRBYT SEI
0820      LDA #$FE
0830      STA $D301
0840      LDA # <RTI
0850      STA NMIVEC
0860      STA IRQVEC
0870      LDA # >RTI
0880      STA NMIVEC+1
0890      STA IRQVEC+1
0900      LDX #0
0910 SKAD LDA TRAM,X
0920 GDZIE STA ROM,X
0930      INX
0940      BNE SKAD
0950      CLI
0960      LDA #253
0970      STA PORTB
0980 RTS RTS
0990 ;
1000 RTI RTI
1010 ;
1020 DATA .BYTE 162,0,160,250,234
1030      .BYTE 234,234,234,234,234
```

SPRZĘT

SYSTEMY instalowane w stacjach dysków

ATARI 1050

Producent: Atari Corp., USA
Procesor: 6507
ROM: 4 KB
RAM: 2 KB — 6810
PIO: 6532
Kontroler: 2793
Transmisja: 19200 bodów
Gęstości: S, E, M

Oprogramowanie systemowe:

Dyskietka systemowa dla stacji 1050 zawiera: DOS 2.5 — opracowany przez Atari Corp., RAMDISK.COM — program instalujący i formatujący ramdysk (D8:) w komputerach 130 XE, SETUP.COM — program do zmiany konfiguracji DOS-u oraz do tworzenia pliku AUTORUN.SYS uruchamiającego wskazany program w Atari Basic, COPY32.COM — program do przenoszenia plików z dyskietek w formacie DOS 3.0, DISKFIX.COM — program do odzyskiwania skazowanych plików, weryfikacji VTOC i zmieniania nazw plików. UWAGA: W Pewexie stacje Atari 1050 były sprzedawane z DOS 3.0 zamiast DOS 2.5.

Oprogramowanie dodatkowe:

Ponieważ Atari 1050 posiada możliwość pracy w formacie rozszerzonym (określanym jako *DUAL, ENHANCED lub MEDIUM)*, można stosować wszystkie dyskowe systemy operacyjne, które mają możliwość pracy w takiej gęstości, np. BIBODOS, SUPERDOS, SPARTADOS, DOS 3.0, MYDOS, DOS II+ oraz systemy pracujące w gęstości pojedynczej, np. DOS 2.Os, DOS XL, SMARTDOS, DOS 4.0 i inne. Monitory dyskowe pracujące w formacie rozszerzonym to: UMON XL, SHERLOCK 1050, WATSON, TURBO WATSON, HACKER KING, DISKOVER, DISK DUMPER XL, HYPRA MONITOR. Zalecane programy kopiujące: SECTOR COPIER 2, SYNCOPY, TRACK COPIER, US COPY 4.

UWAGI:

System operacyjny Atari 1050 nie daje możliwości kopiowania dyskietek zabezpieczanych sektorami typu BAD, DDM, CRC, WEAK oraz zapisanych sektorami asynchronicznymi i synchronicznymi w odpowiedniej kolejności.

DUPLIKATOR 1050

Producent: Duplicating Technologies Ltd., USA
Procesor: 6502
ROM: 8 KB
RAM: 8 KB — 6264
PIO: 6532
Kontroler: 2793
Transmisja: 70000 bodów
Gęstości: S, E, M, D

Oprogramowanie systemowe

Podstawowe oprogramowanie zajmuje 3 dwustronnie zapisane dyskietki w pojedynczej gęstości. Strona 1: THE HIGH SPEED BOOTER — instalacja trybu US (Ultra Speed) w komputerze. Strona 2: MAIN COPY PROGRAM — program kopiujący (gęstość pojedyncza) oraz cztery moduły RPM do kontroli prędkości obrotowej stacji dysków. Strona 3: COPY XE, MOD COPY, SKEW COPY — programy kopiujące dyskietki zabezpieczane sektorami typu BAD, DDM, CRC, DUPLICATE i zapisane z synchronizacją początku ścieżki. (SKEW SECTORS). Strony 4—6: DUPLICATOR SUPPORT SOFTWARE — moduły tworzące odbezpieczone kopie firmowo zabezpieczanych programów takich jak PRINT SHOP, LODE RUNNER itp. Wersja 5.0 oprogramowania kopiuje zabezpieczone programy zawierające do 30 sektorów na ścieżce.

Oprogramowanie dodatkowe:

Specjalne zestawy pod nazwą CHIPMUNK do odbezpieczania i ponownego zabezpieczania firmowych wersji ponad dwustu programów. Poza DOS-em systemowym stosuje się SUPERDOS 4.3E i SPARTADOS, ponieważ są one przystosowane do włączania i pracy w trybie US.

Możliwości:

DUPLICATOR 1050 umożliwia kopiowanie takich programów jak DECISION ON DESERT, CRUSADE IN EUROPE itd. Komunikacja użytkownik — kontroler powiększona została o komendę $3F (włącz tryb US).

Wady systemu:

Kłopoty przy pracy w gęstościach E i M oraz przy przeformatowywaniu z gęstości S na E i M.

Zalety systemu:

Kopiowanie SKEW SECTORS oraz uruchamianie z prędkością US.

TURBO 1050

Inne nazwy: TOP DRIVE 1050, HAPPY 1050, HAPPY WARP, TOP
Producent: Bernhard Engel RFN
Procesor: 6507
ROM: 8 KB
RAM: 2 KB — 6810
PIO: 6532
Kontroler: 2793
Transmisja: 70000 bodów
Gęstości: S, E, M, D

Oprogramowanie systemowe:

Całość oprogramowania systemowego (program kopiujący, edytor i formater ścieżek oraz tester prędkości obrotowej) jest zawarta w pamięci ROM wbudowanej do stacji.

Oprogramowanie dodatkowe:

Poza oprogramowaniem stosowanym w stacjach Atari 1050 stosuje się programy narzędziowe pracującej w podwójnej gęstości (DISK WIZARD II, DISK SCANNER II, TOP ARCHIVER oraz DOS-y posiadające możliwość pracy w tym formacie).

Możliwości:

Dyskietka formatowana w gęstości podwójnej może na jednej stronie zmieścić maksymalnie 180 KB, co daje zysk około 50 KB w stosunku do formatu rozszerzonego. Nie do pogardzenia jest również wzrost prędkości transmisji do 70000 bodów. Ponadto wbudowany program kopiujący kopiuje 95% dyskietek w gęstości pojedynczej zabezpieczanych sektorami BAD, CRC, DDM i DUPLICATE). W komputerach posiadających 128 KB pamięci RAM podczas kopiowania program zakłada ramdysk, co pozwala kopiowanie dyskietek w jednym przebiegu oraz wykonywanie następnych kopii z ramdysku. Korzystając z edytora ścieżek użytkownik może wprowadzić własne zabezpieczenia oraz przeplot sektorów. Istnieje możliwość zainstalowania w stacji wyjścia równoległego CENTRONICS.

Wady systemu:

Poza komendami $4E i $4F brak jest możliwości komunikacji użytkownika z kontrolerem stacji.

Zalety systemu:

Najtańsze rozszerzenie możliwości stacji Atari 1050.

UWAGA:

Ponieważ bardzo wielu użytkowników mylnie stosuje nazwy HAPPY, proponuję porównanie systemu TURBO 1050 z systemem HAPPY WARP 7.10.

Zbigniew GRADOMSKI

Efektem kompilacji programu napisanego w Action! jest program w języku maszynowym. Jeśli podczas jego pisania przestrzegaliśmy zasad podanych w poprzednim odcinku, to program ten będzie działał także bez samego Action!. Taki program maszynowy można dołączyć do innego programu stworzonego w każdym niemal języku programowania. Na przykład, w Turbo Basicu.

Podczas kompilacji **Action!** sam określa adres początkowy programu w pamięci. Istnieje jednak sposób ustalenia tego adresu przez użytkownika. Należy tylko wpisać go do komórek pamięci o adresach $0E i $0491. Najlepiej wykonać to za pomocą dyrektywy SET, użytej nie jako rozkaz monitora, lecz jako dyrektywa kompilatora. Na przykład:

```
SET $E=20000
SET $491=20000

BYTE blablabla
CARD blablablabla

PROC JAKAS_TAM()
```

W ten sposób skompilowany program rozpocznie się od adresu 20000. Pamiętaj, że oba wskazane adresy muszą mieć tą samą wartość. Rejestr $0E zawiera adres kompilacji, a $0491 oznacza początek pamięci zapisywanej poleceniem monitora „W".

Są jeszcze trzy rzeczy, o których należy koniecznie pamiętać łącząc **Action!** z innym językiem programowania:

OBSZAR PAMIĘCI

Skompilowany program w **Action!** musi zajmować obszar pamięci niewykorzystywany przez język, z którym go łączymy. Dolna granica pamięci RAM dostępnej dla użytkownika jest przechowywana w rejestrze MEMLO (komórki 743 i 744), górna zaś w MEMTOP (741 i 742). Zawartość MEMTOP zmienia się i zależy od wybranego trybu graficznego. W obszarze tym musi znaleźć się cały program. Dla programu w **Action!** pozostaje więc obszar między końcem programu w innym języku, a MEMTOP. Na przykład, adres końca programu w **Basicu** lub **Turbo Basicu** jest przechowywany w komórkach 144 i 145 (BMEMH). Od tego adresu do MEMTOP znajduje się obszar, który może być przeznaczony dla **Action!**. BMEMH jednak również ulega zmianie, gdy deklarujemy nowe tablice, wywołujemy procedury lub otwieramy pętle.

Z tych powodów najlepszym rozwiązaniem jest zarezerwowanie dla **Action!** specjalnego obszaru pamięci. Realizuje się to podobnie jak przy rezerwowaniu miejsca dla grafiki graczy i pocisków lub zestawów znaków. Korzysta się w tym celu z rejestru RAMTOP (106), w którym znajduje się numer najwyższej strony pamięci stojącej do dyspozycji. Wystarczy tylko zmienić jego zawartość i wykonać instrukcję GRAPHICS (w celu przesunięcia pamięci obrazu). W **Turbo Basicu** na przykład:

POKE 106, 160: GRAPHICS 0

Teraz mamy do dyspozycji bezpieczny obszar od 40960 do 49152 (160*256 = 40960). Do tego obszaru można już bezpośrednio wczytać program w **Action!**.

Pojawia się tu wszakże kolejny problem: programu w **Action!** nie można skompilować w takim obszarze pamięci, gdyż jest on zajmowany przez moduł **Action!** — jest to możliwe tylko w wersji dyskowej lub kasetowej. Można to jednak rozwiązać korzystając z tzw. kompilacji OFFCODE. Na czym to polega? Program można skompilować w normalnie dostępnym obszarze pamięci, lecz w taki sposób, że będzie on później umieszczany w innym obszarze. Do określenia przesunięcia programu wynikowego służą komórki $B5 i $B6. Musi być w nich wpisana różnica między właściwym adresem umieszczenia programu, a adresem, od którego będzie on kompilowany. Na przykład, program powinien być umieszczony od adresu $A000, ale podczas kompilacji będzie zapisywany od adresu $5000. Trzeba więc na początku programu napisać:

```
SET $E=$5000
SET $491=$5000
SET $B5=$5000
```

gdyż $5000 + $5000 = $A000. Gdy rejestr $B5 nie był ustawiany, to znajduje się w nim zawsze wartość 0.

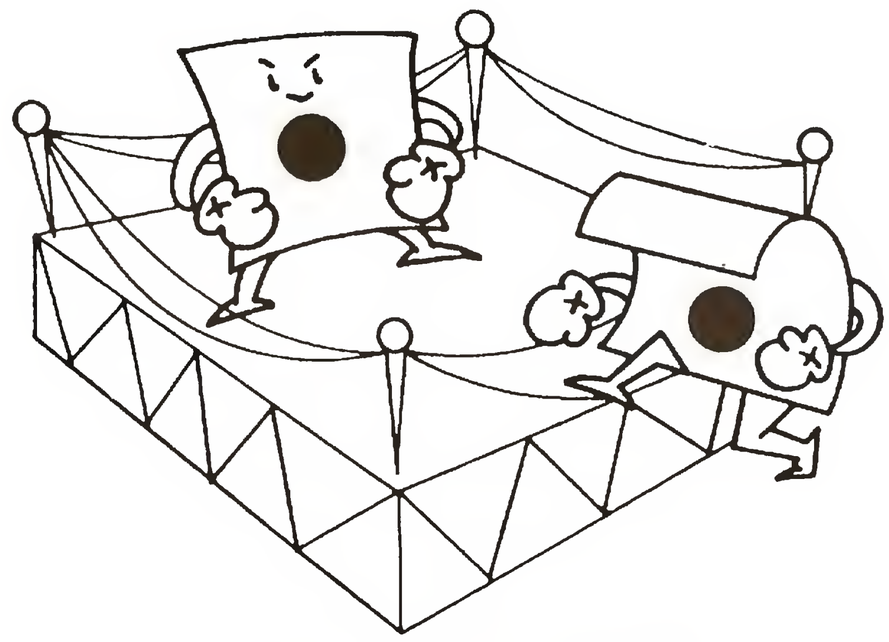
DZIAŁANIE PROGRAMU

Jeżeli nie posiadamy biblioteki OSS Runtime Library, to program w **Action!** będzie działał tylko wtedy, gdy zostanie napisany zgodnie z zasadami opisanymi w poprzednim „Moim Atari" („Action! bez Action!"). Muszą być więc spełnione następujące warunki:

a) program nie wykorzystuje żadnych PROCedur bibliotecznych **Action!**;
b) do każdej procedury nie jest przekazywane więcej niż trzy bajty parametrów (na wszelki wypadek lepiej przekazywać tylko dwa bajty);
c) w programie nie ma żadnego mnożenia i dzielenia;
d) operatory przesunięcia (RSH i LSH) nie są używane z wartościami typu CARD i INT.

URUCHOMIENIE ACTION!

W większości wypadków zapisuje się skompilowany program w **Action!** rozkazem „W" monitora. Na końcu pliku jest wtedy zapisywany jako ostatni segment adres uruchomienia (INIT). Adres ten (komórki 738 i 739) wskazuje, ostatnią PROCedurę w programie. Zapisany w ten sposób program będzie automatycznie uruchomiony po każdym wczytaniu do komputera. W **Turbo Basicu** do wczytania takiego programu trzeba użyć instrukcji BRUN lub BLOAD



— i zawsze nastąpi samoczynne uruchomienie wczytanego programu.

Niestety, nie ma takiej możliwości w **Atari Basic**. Wczytany program trzeba więc uruchamiać instrukcją USR. Pojawia się tu jednak dodatkowy kłopot, gdyż USR zapisuje na stosie liczbę przekazywanych parametrów. Nawet, gdy ich nie ma, zapisywana jest wartość 0. Normalnie więc wszystkie procedury wywoływane przez USR zaczynają się od rozkazu PLA, który zdejmuje ze stosu bajt określający liczbę parametrów. Jest to konieczne dla zapewnienia poprawnej pracy programu. Co w takim przypadku zrobić w **Action!**? Nie ma tam przecież instrukcji PLA.

To żaden kłopot! W **Action!** można dodawać fragmenty programu maszynowego jako bloki kodu. Są to ograniczone nawiasami kwadratowymi liczby odpowiadające rozkazom procesora. Trzeba więc tylko dodać jako blok kodu wartość 104, która odpowiada rozkazowi PLA. Na przykład:

```
PROC XY()
[104]

RETURN
```

Można też oczywiście wpisać jako blok kodu większy fragment programu maszynowego: [104 169 1 ...]. Ponadto w bloku kodu można wpisać zmienne **Action!** zamiast bezpośrednich adresów.

[104] najlepiej jest umieścić na samym początku ostatniej PROCedury. Zapewnia to bezbłędne działanie programu i umożliwia w razie konieczności powrót do **Basica**.

Niezbędna jest jeszcze znajomość adresu początkowego ostatniej PROCedury, aby podać go w instrukcji USR. Adres początkowy, który był użyty w dyrektywie SET i został wpisany do rejestrów $0E i $0491, NIE jest adresem ostatniej procedury. Trzeba więc skompilować program w **Action!**, a następnie odczytać adres ostatniej PROCedury za pomocą rozkazu „?" monitora wpisując:

?nazwa_PROC

Opisane tu możliwości połączenia **Action!** z innymi językami programowania z pewnością umożliwią zwiększenie atrakcyjności programów i ułatwią ich tworzenie.

Wojciech ZIENTARA

Nabywcy dyskietki z programami opublikowanymi w tym numerze „Mojego Atari" otrzymają jako premię dwa programy ilustrujące opisywane tu łączenie **Turbo Basica** i **Action!**. Programy te pochodzą z niemieckiego pisma „Atari Magazin".

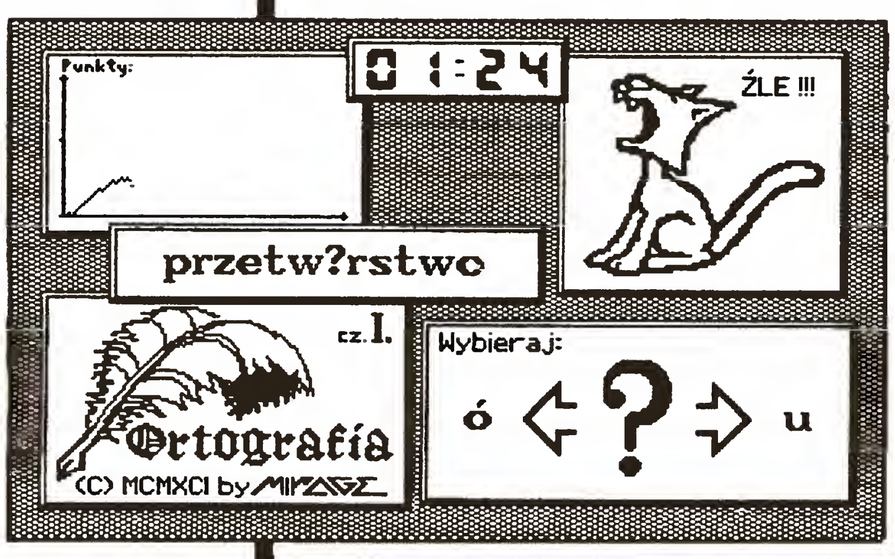
PROGRAMOWANIE

Action! + BASIC

RECENZJE

ORTOGRAFIA

Program „Ortografia" jest pierwszą z serii gier edukacyjnych proponowanych przez Atari Studio, a produkowanych przez Mirage. Pod szyldem tej firmy ukazała się wcześniej gra BATTLE SHIPS oraz krążąca niestety pokątnie (często w mocno zmienionej i okrojonej wersji) WARSAW TETRIS.



„Ortografia" jest programem bardzo atrakcyjnym z wielu względów. W ciekawy sposób połączono w nim elementy zabawy z edukacją. Staranne opracowanie graficzne i dobra muzyka stanowią jedną z zalet. Najważniejsze są jednak inne elementy: słownik programu składa się z ponad 1000 słów, więc hasła do odgadnięcia powtarzają się bardzo rzadko, gracz nigdy nie widzi na ekranie nieprawidłowo napisanego wyrazu (bardzo ważne ze względu edukacyjnego), program poprzez możliwość zapisu osiągniętego wyniku wprowadza element rywalizacji, końcowy wynik zależy nie tylko od poprawności odpowiedzi, ale także od czasu namysłu i ilości odgadniętych wyrazów w określonym czasie (2 minuty), obsługa programu jest niezwykle prosta.

Ekran programu składa się z kilku okien: pytanie, wybór właściwej pisowni, ikona obrazująca poprawność odpowiedzi, wykres zdobywanych punktów, zegar upływającego czasu, informacja o programie. Obsługa odbywa się przy użyciu joysticka, wyboru poprawnej odpowiedzi dokonujemy przesuwając go w lewo lub w prawo.

Gra została bardzo estetycznie wydana — ładna kaseta, kolorowa wkładka, wyczerpująca instrukcja, a w wersji dyskowej kolorowa koperta z podobną wkładką jak dla kasety to na naszym rynku niewątpliwe nowości. Zalety te wraz z niską ceną (proponowana przez dystrybutora cena detaliczna 20000—25000 zł) stanowią o dużej atrakcyjności programu.

Na zakończenie jeszcze parę słów zachęty dla użytkowników i firm zajmujących się sprzedażą oprogramowania. Pomimo, iż prawa autorskie w zakresie oprogramowania są jeszcze „w powijakach", to należy pamiętać, że popierając oficjalnych dystrybutorów i sprzedając oryginalne kopie programów oszczędzają sobie Państwo problemów z wczytywaniem, czy brakami w programie. Przede wszystkim jednak zachęca to autorów do pisania nowych, co przyczynia się do wzbogacenia oferty programowej oraz normalizacji rynku oprogramowania, która może przynieść tylko korzyści zarówno programistom, dystrybutorom, jak i samym użytkownikom komputerów.

Firma Atari Studio (Warszawa, ul. Gen. Abrahama 4, tel. 125-123) jako oficjalny dystrybutor programów MIRAGE Software Ltd. zaprasza do współpracy wszystkich piszących programy, zapewniając godziwe honoraria za wykonaną pracę. Jednocześnie poleca zarówno odbiorcom indywidualnym jak i firmom sprzedającym programy i wydany wcześniej BATTLE SHIPS oraz przygotowywane GEOGRAFIA POLSKI, gry RODERIC, BERTYX, BOMB JANE BAD oraz książkę z kasetą (lub dyskietką) „Co można zrobić z komputerem Atari XL/XE".

(tm)

Program:	**ORTOGRAFIA**
Komputer:	**Atari XL/XE**
Autor:	**Arkadiusz Łukszo**
Muzyka:	**Jakub Husak**
Producent:	**MIRAGE Software Ltd**
Dystrybutor:	**Atari Studio „AS"**

KONKURSIK

Wielki Konkurs!

Zapraszamy na drugie wydanie nieustającego konkursu dla czytelników naszego pisma. Jak poprzednio do wygrania są następujące akcesoria ufundowane przez firmę "TAL"

1 joystick SV 127

10 joysticków SV 120

5 joysticków SV 119

3 wycinarki do dyskietek 5.25"

2 paczki po 10 dyskietek 5.25"

1 pudełko na dyskietki 5.25"

W konkursie może wziąć udział każdy, kto zechce odpowiedzieć na kilka pytań i wypełni dokładnie zamieszczoną niżej ankietę. Nie zależy nam, aby pytania konkursowe były trudne, lecz udzielenie poprawnych odpowiedzi będzie wymagało dokładnego przeczytania bieżącego numeru "Mojego Atari".

Pod każdym pytaniem znajdują się trzy odpowiedzi, lecz tylko jedna z nich jest prawidłowa. Wystarczy ją podkreślić. Ci, którzy nie chcą niszczyć pisma wycinaniem, mogą odpowiedź zapisać na oddzielnej kartce w formie "1-a, 2-c itd.

1 Jaki układ scalony w stacjach dysków LDW i CA jest sterowany przyciskiem PROTECT?

- 74LS02;
- 74LS14;
- 74LS74.

2 Jak nazywa się firma produkująca program "Ortografia"?

- Mirage;
- MicroProse;
- Atari Studio.

3 Jaka stacja dysków umożliwia zapisanie największej liczby danych na dyskietce?

- CA 2001;
- XF 551;
- TOMS 720.

4 Do czego służy pakiet programowy pod nazwą COBRA?

- do zarządzania danymi na dyskach optycznych;
- do sterowania przemysłowym komputerem 190ST020;
- do tworzenia programów symulujących śmigłowce.

5 Jaką nazwę nosi jeden z pierwszych emulatorów Apple na Atari ST?

- Saracen;
- Aladyn;
- Sindbad.

ANKIETA

Imię i nazwisko: .....

.....

dokładny adres: .....

.....

posiadany komputer: Atari 800XL, 65XE, 130XE, 520ST 1040ST, inny: .....

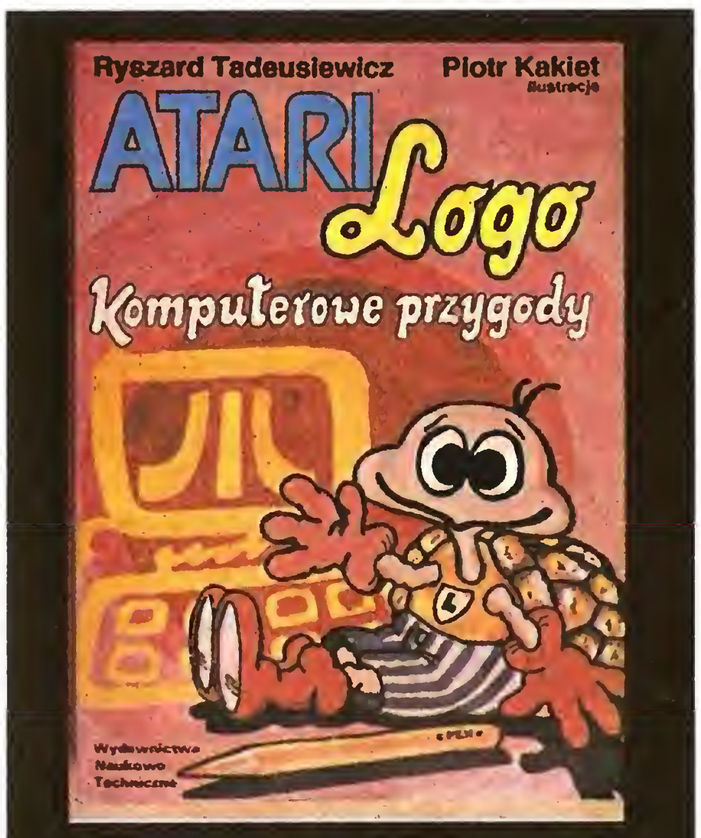
rozmiar pamięci RAM komputera: .....

stosowana pamięć masowa: magnetofon, turbo, stacja dysków.

wiek: ..... lat.

Uwaga!

Odpowiedzi przyjmujemy do dnia ukazania się kolejnego numeru "Mojego Atari".



ATARI LOGO
— komputerowe przygody

LOGO zostało wymyślone w Stanach Zjednoczonych jako język przeznaczony dla szkół do początkowego nauczania informatyki. Zdobyło sobie prawie natychmiast duże powodzenie u dzieci i młodzieży z całego świata, natomiast przez profesjonalistów było i jest nadal często traktowane po macoszemu.

Powodów ku temu podaje się wiele, ale chyba najczęściej pojawia się stwierdzenie, że jest to język zbyt prosty, a nawet momentami prymitywny. Ale czy można to traktować jako wadę? Oczywiście że nie! Przecież zamierzeniem jego autorów było właśnie stworzenie łatwego, a jednocześnie w miarę dobrego języka (decyduje o tym np. jego strukturalność, której tak brakuje Basicowi).

Zalety niezwykłej prostoty LOGO skłoniły profesora Ryszarda Tadeusiewicza do napisania książki „ATARI LOGO — Komputerowe przygody".

Nie jest to kolejny, nudny i zbyt mądry podręcznik programowania, ale napisana w przystępny sposób książka, adresowana do szerokiego grona czytelników. Człowiek, zabierający się do jej czytania, nie przerywa lektury po kilku minutach, gdyż kolejne strony wzbudzają coraz większe zainteresowanie, zachęcając do zabawy i eksperymentowania z LOGO.

Dużą zaletą podręcznika jest to, że autor od początku traktuje czytelnika jako kogoś, kto nie miał jeszcze styczności z komputerem. Dzięki temu nie zauważa się dysproporcji pomiędzy poziomem swoich wiadomości i komputerowego doświadczenia, a wiedzą autora.

Tak więc książkę rozpoczynają elementarne informacje o tym, jak wczytać program LOGO z kasety lub dysku do pamięci. Dalej znajduje się rozdział poświęcony obsłudze klawiatury i podstawowym poleceniom służącym do czyszczenia ekranu, kierowania ruchami żółwia. Potem następuje wprowadzenie bardziej zaawansowanych i złożonych struktur LOGO czyli procedur, sterowania kilkoma żółwiami, rysowania wykresów, skomplikowanych rysunkow, itp. Wszystko to poparte jest przykładami ułatwiającymi zrozumienie i zapamiętanie przekazywanej wiedzy.

Lekturę uprzyjemniają liczne, bardzo zabawne ilustracje autorstwa Piotra Kakieta, które, stwarzają swoisty nastrój, sprawiający, że czytelnik uczy się LOGO z ogromnym zapałem i przyjemnością.

Książka napisana jest w sposób ciekawy, odbiegający od standardów znanych z innych podręczników komputerowych. Zamierzeniem autora było, jak sam tytuł wskazuje, umożliwienie czytelnikowi przeżycia komputerowej przygody. Ułatwia to przyswojenie sobie materiału, a także zachęca do dalszego pogłębiania informatycznych zainteresowań.

Podsumowując, uważam, tę książkę za znakomitą. Polecam ją wszystkim rodzicom, którzy chcieliby zafundować swoim dzieciom interesujący i pożyteczny upominek. „Atari LOGO — komputerowe przygody" jest rownież świetną lekturą dla dorosłych, chcących rozpocząć swoją edukację komputerową. Jedynym mankamentem tego opracowania jest brak dołączonej kasety lub dyskietki z programem LOGO, albo np. z przykładami programów przeznaczonych do przeanalizowania.

Ryszard Tadeusiewicz „ATARI LOGO — Komputerowe przygody", Wydawnictwa Naukowo-Techniczne, Warszawa 1991, Wyd I, 287 stron.

ATARI JAKO STEROWNIK

Zachęcony licznymi publikacjami na temat wykorzystania komputerów do sterowania różnych urządzeń postanowiłem wykonać odpowiedni interfejs do Atari XL/XE. Korzystając ze wskazówek zawartych w artykule o niekonwencjonalnym wykorzystaniu portów joysticków („Bajtek" 4/88) zbudowałem sterownik z układem 74145. Niestety włączane kanały sterujące były zależne i jednocześnie mógł być włączony tylko jeden z nich.

To mnie nie zadowalało. Zwróciłem uwagę na układ 7407, lecz nigdzie nie mogłem go zdobyć. Zastępczo użyłem więc 7406, który ma tą wadę, że sterownik można włączać dopiero po uruchomieniu programu sterującego. Wcześniejsze włączenie sterownika powoduje bowiem stan aktywny wszystkich kanałów sterowania. Układ 7407 zawiera wzmacniacze buforowe, a 7406 — inwertery buforowe i stąd biorą się te niedogodności. Schemat układu sterującego znajduje się na rysunku 1.

Kolejny etap to wykonanie części włączającej napięcie 220 V. Ponieważ część ta musi być oddzielona od komputera, użyłem separacji optoelektronicznej wykorzystując fotorezystory, a w części wykonawczej przekaźniki (rysunek 2). Oczywiście można posłużyć się fotodiodą lub fototranzystorem zmieniając odpowiednio schemat. Zachęcam do szukania w różnych czasopismach poświęconych elektronice schematów wyłączników zmierzchowych i dopasowania ich do posiadanych elementów.

Działanie sterownika polega na tym, że dioda świecąca powoduje poprzez element reagujący na światło włączenie części wysokonapięciowej. Sam układ 7406 jest zasilany z komputera, a diody elektroluminescencyjne z zewnętrznego zasilacza o napięciu 5 V. Część wykonawcza jest zasilana napięciem zależnym od zastosowanych przekaźników i fotoelementów (ja użyłem napięcia 17 V). Dioda w obwodzie przekaźnika zabezpiecza tranzystor przed prądem zwrotnym. Kondensator w obwodzie 220 V zmniejsza iskrzenie na stykach przekaźnika w chwili rozłączania. Przedstawione urządzenie jest czterokanałowe, ale nic nie stoi na przeszkodzie, aby wykorzystać osiem wyjść i stworzyć urządzenie ośmiokanałowe.

Do obsługi sterownika wykorzystałem program zamieszczony w książce W. Zientary „Mapa pamięci Atari XL/XE. Procedury wejścia/wyjścia", a zaczerpnięty z „Atari User". Program ten instaluje nowe urządzenie zewnętrzne oznaczone jako „J:". Urządzenie to może wykonywać instrukcje OPEN, GET, PUT i CLOSE.

OPEN otwiera wybrany kanał IOCB, przy czym możliwe jest użycie następujących parametrów:

OPEN #n,4,0,"J:" — otwiera kanał n do odczytu,

OPEN #n,8,0,"J:" — otwiera kanał n do zapisu,

OPEN #n,12,x,"J:" — otwiera kanał n do zapisu i odczytu (x określa bity wejściowe i wyjściowe).

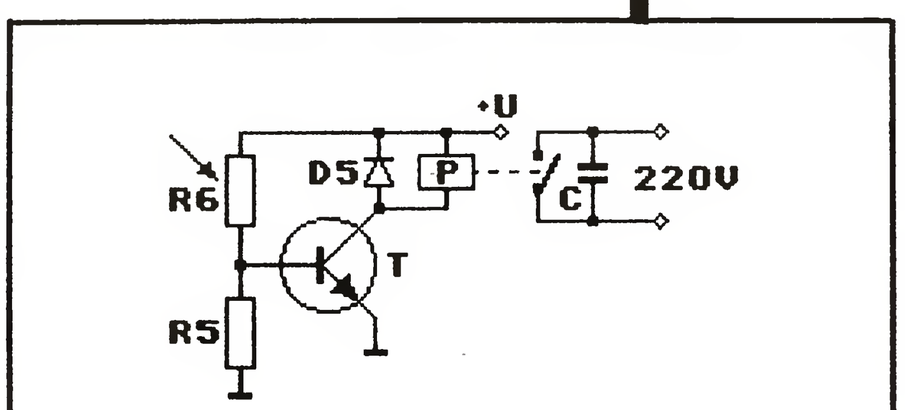
Instrukcja **OPEN #1,12,15,"J:"** otwiera do zapisu port joysticka 1. Poprzez wpisywanie instrukcją PUT wartości z zakresu od 0 do 15 włączamy poszczególne kanały sterowania. Pierwszemu kanałowi odpowiada wartość 1, drugiemu 2, trzeciemu 4 i czwartemu 8. Na przykład w celu włączenia drugiego i trzeciego kanału (i wyłączenia pozostałych) należy podać wartość 2+4=6. Napisanie w Basicu prostego programu dostosowanego do potrzeb użytkownika nie powinno nastręczać trudności.

Marek RUTA

Wykaz elementów

US	— 7406, 7407, 74016, 74017
T	— BC107, BC108, BC147, BC148
D1-D4	— dowolne LED
D5	— BYP150, BYP401, BAP795
C	— 470 nF lub więcej na 400 V
R1-R4	— 220-330 Ω (dobrać zależnie od D1-D4)
R5	— 1 kΩ (dobrać do przekaźnika)
R6	— RPP130, RPP120 lub inny
P	— przekaźnik R15 lub inny

Rys. 1



Rys. 2

```
NN   900 S=0:FOR X=1 TO 113:READ A:S=S+A:PO
     KE 1535+X,A:NEXT X
BT   910 IF S<>10694 THEN ? "BLAD W DATA":E
     ND
YS   920 A=USR(1536)
AP   1000 DATA 104,162,0,189,26,3,240,11,23
     2,232,232,224,34,176,3,76,3,6
YI   1010 DATA 0,169,74,157,26,3,232,169,37
     ,157,26,3,232,169,6,157,26,3
NG   1020 DATA 96,51,6,96,6,105,6,99,6,111,
     6,111,6,76,86,6,165,42,201,12
QU   1030 DATA 240,11,201,8,240,25,201,4,24
     0,26,160,146,96,166,43,169,56
NX   1040 DATA 141,2,211,142,0,211,169,60,1
     41,2,211,160,1,96,162,255,76,71,6
UD   1050 DATA 162,0,76,71,6,76,92,6,141,0,
     211,76,84,6,173,0,211,76,84,6,96
```

SZKÓŁKA

FORMATY DYSKIETEK

W poprzednim numerze „Mojego Atari" poznaliśmy budowę dyskietki stosowanej do zapisu danych komputerowych. Pora teraz przejść do opisu sposobu, w jaki dane te są umieszczane na dyskietce.

Wyobraź sobie zwykłą, czystą kartkę papieru. Aby na niej cokolwiek zapisać, wystarczy wziąć długopis lub ołówek i pisać. Tak możesz zrobić Ty, lecz komputer jest od Ciebie znacznie głupszy. Na czystej dyskietce nic nie zapisze, bo nie wie, w którym miejscu. Przed użyciem dyskietka musi być odpowiednio przygotowana — operacja taka nazywa się formatowaniem.

Na czym polega formatowanie dyskietki? Wyobraź sobie przez chwilę, że komputer nie zapisuje na dyskietce, lecz na kartce. Aby mu umożliwić zapis i odczyt trzeba z tej kartki zrobić formularz. W tym celu rysujemy na czystej kartce rubryki: kolumny i wiersze. Teraz komputer może na niej pisać i czytać ją podając stacji numer wiersza i kolumny. Dyskietka jest jednak okrągła i obraca się. Nic prostszego: zwińmy kartkę, aby utworzyć z niej walec w taki sposób, aby koniec każdego wiersza stykał się z jego początkiem. Następnie walec ten trzeba jeszcze spłaszczyć i otrzymujemy koło podzielone w sposób pokazany na rysunku.

Dokładnie tak samo wygląda komputerowa dyskietka. Inny jest tylko materiał, z którego jest wykonana, a wiersze i kolumny są nazywane ścieżkami i sektorami, przygotowanie ich na dyskietce nosi zaś nazwę formatowania. W odróżnieniu od kartki papieru, dyskietkę można formatować wielokrotnie. Często korzysta się z tego sposobu do dokładnego usunięcia zbędnych już danych lub programów i przygotowania dyskietki do zapisu nowych.

Oczywiście, nie dzielimy dyskietki na rubryki samodzielnie — robi to za nas DOS (Dyskowy System Operacyjny), który nadzoruje współpracę komputera ze stacją dysków. Wystarczy tylko wydać polecenie FORMAT lub INIT (w różnych DOS-ach stosuje się różne nazwy). Ponieważ operacja taka niszczy całą dotychczasową zawartość dyskietki, to DOS upewnia się jeszcze, czy na pewno chcemy to zrobić i po potwierdzeniu samo dzielnie formatuje dyskietkę, czy zaznacza na niej ścieżki i sektory.

Byłoby to proste, gdyby nie stały postęp techniczny. W miarę rozwoju techniki rośnie gęstość upakowania danych na dyskietce, czyli przybywa ścieżek i sektorów na nich lub też sektory mieszczą większą liczbę danych. Dzięki temu na tej samej powierzchni dyskietki mieści się więcej informacji. Postęp ten jest tak szybki, że w każdym typie komputera stosuje się obecnie kilka różnych sposobów zapisu. Mówimy więc, że dyskietki są formatowane w różnych gęstościach, a w skrócie, że mają różne formaty. Niektórych użytkowników przyprawia to o ból głowy, więc musimy się teraz zająć tym zagadnieniem.

Początkowo na dyskietkach do komputerów Atari stosowano zapis w gęstości nazwanej później pojedynczą (*single*). W gęstości tej dyskietka jest podzielona na 40 ścieżek, na których z kolei znajduje się po 18 sektorów. W każdym sektorze mieści się 128 bajtów informacji. Razem więc na jednej stronie dyskietki można umieścić 40×18×128=92160 bajtów, czyli 90 KB. Z tak sformatowanych dyskietek korzystała stacja Atari 810. Posiadała ona jedną głowicę, więc zapisanie drugiej strony dyskietki wymagało jej odwrócenia, lecz w ten sposób można było zmieścić dwukrotnie więcej informacji. Oczywiście pod warunkiem, że jakość dyskietki na to pozwalała, co początkowo zdarzało się raczej rzadko.

Bardzo szybko wprowadzono jednak ulepszony sposób zapisu na dyskietce, co pozwoliło na zwiększenie jej pojemności. Dokonano tego poprzez dwukrotne zwiększenie liczby bajtów w sektorze (do 256). W takiej gęstości — zwanej podwójną (*double*), a czasem prawdziwą podwójną (*true double*) dla odróżnienia od średniej (patrz niżej) — na jednej stronie dyskietki można zmieścić 40×18×256=184320 bajtów, czyli 180 KB. Do takich dyskietek przeznaczona była stacja Atari 815.

Już wkrótce okazało się jednak, że krok ten został wykonany zbyt szybko. Jakość ówcześnie produkowanych dyskietek nie gwarantowała bowiem poprawnego zapisu informacji. W ten sposób Atari 815 bardzo szybko przeszła do muzeum, a zastąpiła ją Atari 1050. W tej stacji zastosowano rozwiązanie pośrednie: zamiast zagęszczać bajty w sektorze, zwiększono liczbę sektorów na ścieżce z osiemnastu do 26. W efekcie pojemność wzrosła do 40×26×128=133120 bajtów, czyli 130 KB. Taki format otrzymał nazwę rozszerzonego (*enhanced*) lub średniego (*medium*), choć firma Atari nadal nazywała go podwójnym (*dual*).

Poza zwiększeniem pojemności dyskietki, największą korzyścią z zastosowania rozszerzonej gęstości była możliwość odczytania części tak zapisanej dyskietki przez DOS-y, które rozpoznawały tylko pojedynczą gęstość. W wypadku gęstości podwójnej nie było to bowiem możliwe. Niestety, okazało się wkrótce, że to posunięcie również było błędne, gdyż w międzyczasie poprawiła się jakość dyskietek. W powstałą lukę weszły inne firmy oferując do komputerów Atari stacje dysków pracujące we wszystkich trzech gęstościach.

W takiej sytuacji firma Atari zareagowała skonstruowaniem nowej stacji XF551. Zastosowano w niej dwie ważne modyfikacje (spotykane już wcześniej w urządeniach innych producentów). Pierwszą z nich jest użycie dwóch głowic, dzięki czemu wyeliminowano konieczność przekładania dyskietki w celu wykorzystania jej drugiej strony. Sama dyskietka nie zmieniła się: na jednej stronie nadal zapisuje się 40 ścieżek po 18 sektorów 256-bajtowych. Dla celów reklamowych nazwano to jednak gęstością poczworną (*quadruple*). Z punktu widzenia użytkownika można rzeczywiście mówić o nowej gęstości, gdyż tylko nieliczne DOS-y potrafią ją rozpoznać.

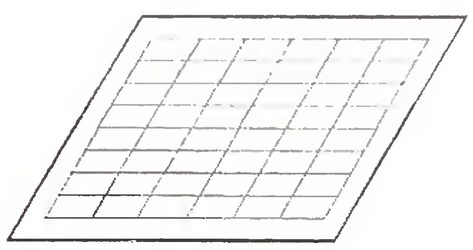
Powtarzające się kłopoty z formatami dyskietek stosowanymi w Atari oraz różnorodność stosowanych stacji dysków skłoniła niektórych producentów oprogramowania do opracowania bardziej uniwersalnych DOS-ów. Należą do nich przede wszystkim SpartaDOS i MYDOS. Nie cieszą się one w Polsce popularnością, gdyż nie znalazły odpowiedniego zastosowania. To z kolei zostało spowodowane bardzo późnym wejściem Atari na nasz rynek — w tym czasie zachodni producenci urządzeń peryferyjnych przestawiali się już na bardziej przyszłościowy sprzęt. Obecność takich systemów spowodowała jednak odwrotną reakcję polskich firm.

Wykorzystując typowe napędy dyskowe przeznaczone do komputerów IBM dwaj warszawscy rzemieślnicy skonstruowali stację TOMS 720 zapisującą dyskietki w gęstości, która rzeczywiście jest poczwórna. Uzyskano to przez dwukrotne zwiększenie liczby ścieżek (z 40 do 80). Stacja ta zapisuje więc na jednej stronie dyskietki 80×18×256=368640 bajtów, czyli 360 KB. Ponieważ TOMS 720 posiada dwie głowice, to faktyczna pojemność dyskietki (uzyskiwana bez jej przekładania) wynosi 720 KB.

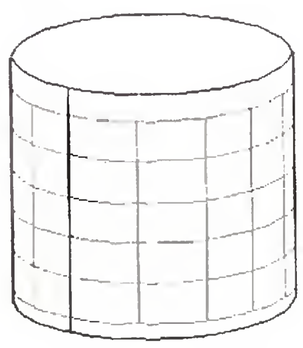
Inną drogą poszła natomiast firma ATARES modyfikując stację CA-2001. System FLASH monto-

FORMAT	SS/SD	SS/ED	SS/DD	DS/DD	SS/TD	DS/QD	S-9	D-9
liczba stron	1	1	1	2	1	2	1	2
liczba ścieżek na stronie	40	40	40	40	40	80	40	40
liczba sektorów na ścieżce	18	26	18	18	6	18	9	9
liczba bajtów w sektorze	128	128	256	256	1024	256	512	512
pojemność obliczeniowa	92160 bajtów	133120 bajtów	184320 bajtów	368640 bajtów	245760 bajtów	737280 bajtów	184320 bajtów	368640 bajtów
pojemność rzeczywista	92160 bajtów	133120 bajtów	183936 bajtów	368256 bajtów	243072 bajty	736896 bajtów	184320 bajtów	368640 bajtów

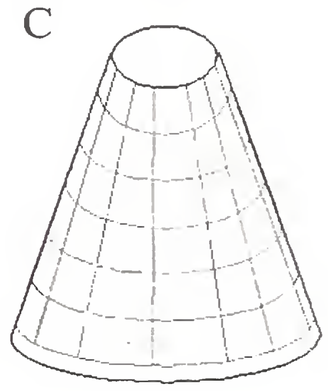
A



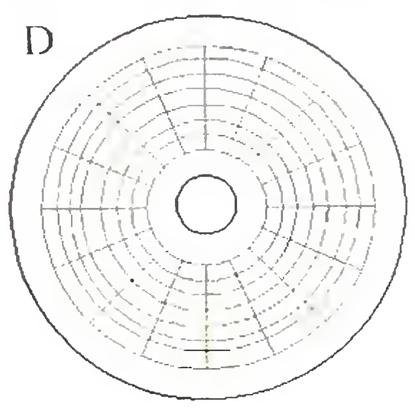
B



C



D



wany w tej stacji — oprócz trzech zwykłych formatów Atari — pozwala także na zapis w swojej specyficznej gęstości nazwanej potrójną. Zasadnicza zmiana polega tu na zmniejszeniu liczby sektorów i zwiększeniu ich wielkości. Tak sformatowana dyskietka posiada 40 ścieżek po 6 sektorów, z których każdy mieści 1024 bajty. Razem można więc zapisać na jej jednej stronie 245760 bajtów, czyli 240 KB.

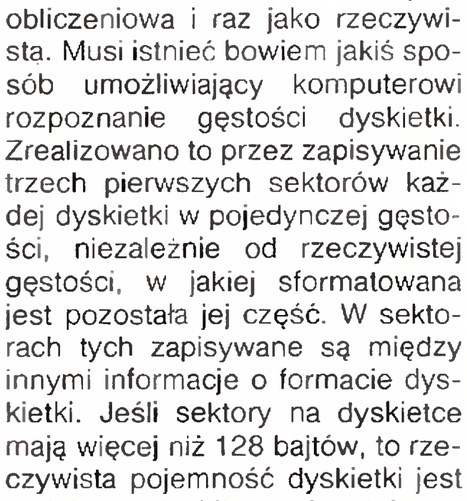
Taka liczba dostępnych formatów rzeczywiście może sprawiać kłopoty. Na dodatek to jeszcze nie koniec. Większość modyfikacji dokonywanych w stacjach dysków umożliwia także rozpoznanie dwóch podstawowych formatów stosowanych w komputerach IBM (S-9 i D-9). Warto więc o nich wspomnieć, choć żaden DOS Atari nie może z tych gęstości korzystać, a wykorzystywane są tylko do przenoszenia danych do i z IBM-a. Format S-9 zawiera na jednej stronie 40 ścieżek po 9 sektorów 512-bajtowych, a D-9 ma identyczny układ sektorów, lecz zapisywany jest na obu stronach dyskietki.

Zorientowanie się w tej gęstwinie gęstości może być trudne, szczególnie dla początkujących użytkowników komputerów. Na zakończenie przedstawmy je więc w formie tabeli. Użyte w niej skróty mają następujące znaczenie:

SS — jednostronny (*Single Sided*),
DS — dwustronny (*Double Sided*),
SD — gęstość pojedyncza (*Single Density*),
ED — gęstość rozszerzona (*Enhanced Density*),
DD — gęstość podwójna (*Double Density*),
TD — gęstość potrójna (*Triple Density*),
QD — gęstość poczwórna (*Quadruple Density*),

Pojemność dyskietki jest podana w tabeli dwukrotnie — raz jako obliczeniowa i raz jako rzeczywista. Musi istnieć bowiem jakiś sposób umożliwiający komputerowi rozpoznanie gęstości dyskietki. Zrealizowano to przez zapisywanie trzech pierwszych sektorów każdej dyskietki w pojedynczej gęstości, niezależnie od rzeczywistej gęstości, w jakiej sformatowana jest pozostała jej część. W sektorach tych zapisywane są między innymi informacje o formacie dyskietki. Jeśli sektory na dyskietce mają więcej niż 128 bajtów, to rzeczywista pojemność dyskietki jest mniejsza od obliczonej, gdyż trzy pierwsze sektory są mniejsze.

Wojciech ZIENTARA



VADEMECUM ATARI

CRUSADE IN EUROPE

„Crusade in Europe" jest grą strategiczną opartą na wydarzeniach kampanii zachodnioeuropejskiej. Kampania ta obejmująca działania wojenne we Francji, Belgii i Holandii w drugiej połowie 1944 roku była jednym z najważniejszych etapów II wojny światowej. Gra przeznaczona jest dla jednego lub dwóch graczy.

KODY ROZPOZNAWCZE

1. TORCH
2. GARDEN
3. GOODWOOD
4. BOLERO
5. MARKET
6. FRANKLIN
7. OMAHA
8. COBRA
9. DAVID
10. TANGO
11. ANVIL
12. TIGER
13. TOTALIZE
14. OVERLORD
15. SHERMAN
16. BLUE COAT

SCENARIUSZE

1. **BATTLE FOR NORMANDY** — lądowanie w Normandii
   początek: 6.06.1944, godz. 6:00

   Warianty:

   1. D-Day: Clearing the Beaches
      Oczyszczenie plaż — historyczny
      koniec: 12.06.1944, godz. 18:00
   2. D-Day: Rommel's Strategy
      Strategia Rommla — hipotetyczny
      koniec: 12.06.1944, godz. 18:00
   3. Breakout from the Beachhead
      Uderzenie z wybrzeża — historyczny
      koniec: 27.06.1944, godz. 18:00
   4. German Quick Reaction
      Szybka reakcja Niemców — hipotetyczny
      koniec: 6.07.1944, godz. 18:00
   5. Liberation of Paris
      Wyzwolenie Paryża — historyczny
      koniec: 25.08.1944, godz. 18:00

2. **RACE FOR THE RHINE** — wyścig do Renu
   początek: 25.08.1944, godz. 12:00

   Warianty:

   1. Short Game
      Krótka gra — historyczny
      koniec: 30.09.1944, godz. 18:00
   2. Long Game
      Długa gra — historyczny
      koniec: 30.10.1944, godz. 18:00

3. **OPERATION MARKET-GARDEN: „A BRIDGE TO FAR"** — Operacja Market-Garden: „O jeden most za daleko"
   początek: 17.09.1944, godz. 13:00

   Warianty:

   1. Historical Situation
      Sytuacja historyczna — historyczny
      koniec: 27.09.1944, godz. 18:00
   2. „Drive on the Ruhr"
      Marsz do Ruhry — hipotetyczny
      koniec: 27.09.1944, godz. 18:00

INSTRUKCJE

4. **BATTLE OF THE BULGE** — kontrofensywa w Ardenach
początek: 16.12.1944, godz. 5:00

Warianty:

1. Dash for the Meuse
Uderzenie nad Mozę — historyczny
koniec: 28.12.1944, godz. 18:00
2. The Full Historical Offensive
Pełna ofensywa historyczna — historyczny
koniec: 6.01.1945, godz. 18:00
3. The „Small Slam"
„Zatrzaśnięcie wieka" — hipotetyczny
koniec: 6.01.1945, godz. 18:00
4. Hitler's Dream
Marzenie Hitlera — hipotetyczny
koniec: 6.01.1945, godz. 18:00

5. **CRUSADE: THE BATTLE FOR FRANCE**
— Krucjata: bitwa o Francję
początek: 6.06.1944, godz. 6:00
koniec: 30.10.1944, godz. 18:00

STEROWANIE

CANCEL — odwołanie rozkazu	
ruch (MOVE)	— ‹M›
atak (ATTACK)	— ‹A›
obrona (DEFEND)	— ‹D›
rezerwa (RESERVE)	— ‹R›
cel oddziału	— ‹H›
mapa terenu	— ‹T›
mapa ogólna	— ‹O›
aktualny wynik gry	— ‹SHIFT›‹?›
zapisanie gry	— ‹S›
odczytanie gry	— ‹L›
zatrzymanie gry	— ‹F›
przyspieszenie gry	— ‹INSERT›
zwolnienie gry	— ‹CLEAR›
powtórzenie gry	— ‹B›
zmiana aktywnego gracza	— ‹Q›
zmiana symboli oddziałów	— ‹U›

MELDUNKI

WE ARE IN CONTACT WITH ENEMY FORCES — jesteśmy w kontakcie z siłami nieprzyjaciela

WE HAVE REACHED OUR OBJECTIVE! AWAITING FURTHER ORDERS — osiągnęliśmy cel, oczekujemy dalszych rozkazów

WE ARE ATACKING, ENEMY IS IN xxx FORMATION, OUR LOSSES ARE HEAVY (MODERATE, LIGHT, VERY LIGHT) — atakujemy, nieprzyjaciel jest w formacji xxx, nasze straty są ciężkie (średnie, małe, bardzo małe)

ENEMY RESISTANCE TOO STRONG, ATTACK MUST BE HALTED — opór nieprzyjaciela zbyt silny, atak musi być wstrzymany

WE HAVE CAPTURED xxxxx — zdobyliśmy xxxxx

WE ARE RETREATING — wycofujemy się

WE HAVE BEEN OVERRUN — zostaliśmy rozbici

MUST SURRENDER — musicie nas ocalić

WE HAVE EXHAUSTED OUR SUPPLIES — zużyliśmy swoje zaopatrzenie

Doświadczenie

RAW	— surowy
GREEN	— zielony
VETERAN	— weterani
CRACK	— wyborowy
ELITE	— elitarny

Formacja

TRANSPORT	— transportowa
MOBILE	— marszowa
DEPLOYED	— rozproszona
DEFENSE	— obronna
ENTRECHEND	— umocniona
FORTIFIED	— ufortyfikowana
GARRISON	— garnizonowa

Zaopatrzenie

AMPLE	— obfite
SUFFICIENT	— dostateczne
CRITICAL	— krytyczne

Pogoda

CLEAR	— pogodnie
OVERCAST	— pochmurnie
FOG	— mglisto
RAIN	— deszcz
SNOW	— śnieg
SLEET	— deszcz ze śniegiem

SYMBOLE JEDNOSTEK

symbole	typ	wielkość	strona
	piechota	dywizja	Alianci
	piechota	brygada	Alianci
	pancerna	dywizja	Alianci
	pancerna	brygada	Alianci
	spadochr.	dywizja	Alianci
	spadochr.	brygada	Alianci
	kaw. panc.	pułk	Alianci
	myśl.-bomb.	skrzydło	Alianci
	piechota	dywizja	Niemcy
	piechota	brygada	Niemcy
	w. fort	dywizja	Niemcy
	grenad.	dywizja	Niemcy
	spadochr.	dywizja	Niemcy
	spadochr.	brygada	Niemcy
	gren. panc.	dywizja	Niemcy
	pancerna	dywizja	Niemcy
	pancerna	brygada	Niemcy
SS	panc. SS	dywizja	Niemcy
	sztab	—	obie
	baza zaop.	—	obie

WYNIKI

Zwycięstwo/klęska

SLIGHT	— pozorne
MARGINAL	— marginalne
TACTICAL	— taktyczne
DECISIVE	— decydujące
TOTAL	— totalne

Stopień

PRIVATE	— szeregowy
SERGEANT	— sierżant
LIEUTNANT	— porucznik
CAPTAIN	— kapitan
MAJOR	— major
LIEUTNANT-COLONEL	— podpułkownik
COLONEL	— pułkownik
BRIGADIER-GENERAL	— generał brygady
MAJOR-GENERAL	— generał dywizji
LIEUTNANT-GENERAL	— generał broni
FIELD MARSHALL	— marszałek
SUPREME COMMANDER	— wódz naczelny

ZAPIS

Jednym z najczęściej używanych programów graficznych jest „Koala Microllustrator". Program ten zapisuje rysunki w specjalny sposób. Prezentuję więc dwa podprogramy (a właściwie procedury w języku maszynowym) służące do zapisywania i odczytywania grafiki 8+16 lub 15 w skompresowanym formacie „Koali".

Pierwsza z prezentowanych procedur kompresuje rysunki zapisując je w standardzie pionowej kompresji używanej przez „Koalę". Pozwala to na zmniejszenie objętości danych średnio do 40–60 procent. Oczywiście odczyt takiego pliku jest możliwy za pomocą specjalnej procedury (np. tej opisanej niżej).

Właściwa procedura zapisu, którą można wykorzystywać zawarta jest w listingu 1. Przed wywołaniem tego podprogramu musi być zadeklarowana zmienna KOAL$ (**DIM KOAL$(340)**), w której będzie przechowywana relokowalna część procedury. Druga część zajmuje 140 bajtów szóstej strony ($600–$68C), o czym należy pamiętać pisząc programy wykorzystujące tą procedurę. Jej użycie jest bardzo proste, a ilustruje je przykładowy program zawarty w listingu 2. Zapisuje on na dyskietce (musi ona być umieszczona w stacji numer 1) przykładowy obraz w formie skompresowanej pod nazwą „PRZYKLAD.PIC".

W celu użycia procedury wystarczy otworzyć dowolny kanał do zapisu, po czym wywołać procedurę instrukcją

I=USR (ADR(KOAL$), kanał*16)

gdzie kanał jest numerem otwartego do zapisu kanału. Po zakończeniu operacji należy kanał ten zamknąć. Jeśli w procesie zapisu wystąpił błąd, to po zakończeniu procedury zmienna I będzie zawierać jego numer, gdy zaś wszystko było w porządku będzie ona równa 1. Tak prosta obsługa zewnętrzna odbyła się kosztem trochę bardziej skomplikowanej budowy procedury w języku wewnętrznym. Źródłowy listing tej procedury jest zawarty na dyskietce, którą można nabyć w redakcji (patrz str. 2). Jest ona napisana w asemblerze MAC/65. Nie będę jej omawiał bowiem łatwo ją przeanalizować.

Teraz kilka uwag (odnoszą się one do obydwu procedur):

1. Po wywołaniu procedury komputer trochę „będzie się zastanawiał" (trwa to około 3–5 sekund). Spowodowane jest to kompresją danych oraz tym, że procedura w rzeczywistości wykonuje się dwa razy! Dzięki temu jednak nie jest potrzebna żadna dodatkowa wolna pamięć. Jest to bardzo duża zaleta tej procedury.
2. Pierwotnie program był uruchamiany tylko na stacji dysków. Nie miałem możliwości sprawdzenia jego pracy z magnetofonem. Powinien jednak pracować z nim bez zastrzeżeń (chodzi o czas przerw pomiędzy blokami).
3. Jeśli listing pierwszy został przepisany bez błędów, to bardzo wygodnie jest zamienić cyfry z instrukcji DATA bezpośrednio na wartości zmiennej KOAL$. Ale to jest już bardzo proste. Zwracam tylko uwagę, że zmienna KOAL$ zawiera znak o wartości 155 na pozycji 103. Taka zamiana skraca kilkanaście razy czas inicjacji, który w podstawowej wersji wynosi około 10 sekund.

KOALI

I ODCZYT RYSUNKÓW

Zasada działania procedury ładującej obrazki „Koali" jest bardzo podobna do opisanej wyżej procedury zapisu. Podam więc tylko istotne różnice.

Procedurę odczytu przedstawia listing 3. Dla tej procedury wymagana jest deklaracja zmiennej OKOAL$ (**DIM OKOAL$ (220)**). Procedura zajmuje również 120 pierwszych bajtów na 6 stronie ($600–$678). Jej obsługa jest prawie identyczna z poprzednią procedurą zapisu i łatwo dokładnie ją przeanalizować na podstawie listingu 4, który zawiera przykładowy program jej wykorzystania. Odczytuje on zbiór utworzony przez przykładowy program zapisu.

Oprócz wyżej podanych trzech uwag należy jeszcze podać trzy następne:

— procedura odczytuje tylko obrazki o kompresji pionowej, ewentualna sprzeczność sygnalizowana jest błędem nr 254;

— jeśli usiłujemy odczytać zbiór, który nie jest zbiorem „Koali", to wystąpi błąd nr 225;

— dodatkowa cyfra (0 lub 1) przy wywołaniu tej procedury pozwala na uniknięcie zmiany kolorów przy odczycie (0 — bez zmiany, 1 — dane dla kolorów są pobierane z odczytywanego pliku).

Program źródłowy procedury odczytu zawarty jest również na dyskietce. Oczywiście jest on także opatrzony komentarzami.

Wypada mi na koniec życzyć szerokich zastosowań i pożytku z obydwu procedur.

Krzysztof KLIMCZAK

LISTING 1

```
XQ   0 REM ZAPIS OBRAZU KOALA
AC   1 REM Krzysztof Klimczak
DU   2 REM (c) 1991, Sp. Bajtek
NI   3 REM
WX   5000 REM Najpierw musi byc zadeklaro-
JR   5002 REM wana tablica KOAL$(340).
FW   5004 REM Procedura wykorzystuje takze
JI   5006 REM adresy 1536-1676 ($600-$68C)
FU   5008 REM Przed wywolaniem trzeba
TX   5010 REM otworzyc kanal do zapisu.
FJ   5012 REM Wywolanie:
WW   5014 REM I=USR(ADR(KOAL$),kanal*16).
AR   5016 REM Po zakonczeniu zamknac kanal.
AT   5020 REM Zwraca wartosc I rowna:
JU   5022 REM 1 - gdy wszystko w porzadku
EX   5024 REM lub numerowi bledu, ktory
YO   5026 REM wystapil przy zapisie.
JM   5028 REM
KB   5040 RESTORE 5100
OW   5045 FOR T=0 TO 6:S=0:FOR T1=1 TO 20:R
     EAD A:S=S+A:POKE 1535+T1+T*20,A:NEXT T
     1:READ A:IF S<>A THEN 5070
JZ   5050 NEXT T
AE   5055 FOR T=7 TO 23:S=0:FOR T1=1 TO 20:
     READ A:S=S+A:KOAL$(T1+(T-7)*20)=CHR$(A
     ):NEXT T1:READ A:IF S<>A THEN 5070
KC   5060 NEXT T
BP   5065 RETURN
UH   5070 ? :? :? "Blad danych w wierszu ";
     T+5100:? :? :? :LIST 5100+T:STOP
IB   5100 DATA 160,0,177,203,72,24,165,203,
     105,80,133,203,144,2,230,204,198,207,2
     08,43,2761
NC   5101 DATA 169,96,133,207,165,209,240,2
     0,198,209,230,205,208,2,230,206,165,20
     5,133,203,3433
AW   5102 DATA 165,206,133,204,198,208,208,
     15,230,209,24,165,205,105,40,133,203,1
     65,206,105,3127
GD   5103 DATA 0,133,204,104,96,133,217,166
     ,218,169,217,157,68,3,169,0,157,69,3,1
     69,2452
TI   5104 DATA 1,157,72,3,169,0,157,73,3,16
     9,11,157,66,3,32,86,228,48,1,96,1532
RB   5105 DATA 132,212,169,0,133,213,166,22
     0,154,96,230,212,208,2,230,213,96,166,
     214,208,3274
LJ   5106 DATA 3,76,110,6,76,65,6,192,0,40,
     0,14,1,1,0,26,199,201,126,255,1399
KQ   5107 DATA 104,186,232,232,134,220,165,
     212,24,105,21,170,165,213,105,0,72,138
     ,72,24,2594
YK   5108 DATA 144,96,104,104,133,218,169,1
     ,133,214,169,12,133,219,166,219,189,12
     7,6,32,2588
QA   5109 DATA 117,6,198,219,16,244,230,219
     ,166,219,189,196,2,224,3,208,2,41,0,32
     ,2531
IP   5110 DATA 117,6,230,219,165,219,201,5,
     208,234,165,212,56,233,1,8,32,117,6,18
     5,2599
MM   5111 DATA 213,40,233,0,32,117,6,169,0,
     32,117,6,169,0,32,117,6,169,4,133,1595
ZE   5112 DATA 219,169,155,32,117,6,198,219
     ,208,247,169,162,32,117,6,24,144,8,169
     ,0,2401
NQ   5113 DATA 133,212,133,213,133,214,169,
     0,133,209,169,96,133,207,169,40,133,20
     8,165,88,2957
SX   5114 DATA 133,203,133,205,165,89,133,2
     04,133,206,32,0,6,133,216,169,1,133,21
     5,165,2674
QQ   5115 DATA 208,240,111,32,0,6,197,216,2
     08,37,230,215,48,29,165,208,240,96,32,
     0,2518
MF   5116 DATA 6,197,216,240,241,166,216,13
     3,216,138,72,165,215,32,117,6,104,32,1
     17,6,2635
GI   5117 DATA 24,144,208,198,215,208,239,7
     2,166,216,133,216,138,72,162,6,181,203
     ,72,202,3075
ZG   5118 DATA 16,250,230,215,48,17,165,208
     ,240,11,32,0,6,197,216,8,133,216,40,20
     8,2456
HL   5119 DATA 237,162,0,104,149,203,232,22
     4,7,208,248,165,215,24,105,127,32,117,
     6,104,2669
WT   5120 DATA 32,117,6,166,215,104,224,2,2
     40,33,208,4,208,137,240,35,32,117,6,19
     8,2324
AQ   5121 DATA 215,198,215,198,215,165,215,
     240,11,32,0,6,32,117,6,198,215,24,144,
     241,2687
HX   5122 DATA 32,0,6,133,216,165,208,208,2
     19,240,10,165,215,32,117,6,165,216,32,
     117,2502
DQ   5123 DATA 6,165,214,240,7,162,1,134,21
     2,202,134,213,96,0,83,77,65,82,75,0,21
     68
QI   5124 REM ****
```

LISTING 2

```
PD   0 REM DEMONSTRACJA ZAPISU KOALA
AC   1 REM Krzysztof Klimczak
HQ   2 REM (c) 1991, SP. Bajtek
NI   3 REM
TI   10 DIM KOAL$(340):REM ta deklaracja mu
     si wystapic
IO   20 GOSUB 5000:REM inicjacja procedury
     maszynowej (okolo 10 s.)
ZW   25 POKE 764,255:? :? "Wloz dyskietke.
     Wcisnij klawisz..."
VG   26 IF PEEK(764)=255 THEN 26
AX   30 GRAPHICS 8+16:COLOR 1:PLOT 0,0:DRAW
     TO 319,0:DRAWTO 0,191:DRAWTO 319,191:D
     RAWTO 0,0:DRAWTO 319,96
SN   40 OPEN #1,8,0,"D:PRZYKLAD.PIC":REM po
     d taka nazwa zapisany bedzie obrazek
UM   50 I=USR(ADR(KOAL$),1*16):REM wywolani
     e procedury
OF   60 CLOSE #1:IF I<>1 THEN ? :? "podczas
      zapisu wystapil blad nr. ";I:STOP
OA   70 ? :? "Uszystko w porzadku.":END
BL   80 REM ***
```

LISTING 3

```
VF   0 REM ODCZYT OBRAZU KOALA
AC   1 REM Krzysztof Klimczak
DU   2 REM (c) 1991, Sp. Bajtek
NI   3 REM
JN   5200 REM Najpierw trzeba zadeklarowac
MG   5201 REM tablice OKOAL$(220).
FS   5202 REM Procedura wykorzystuje takze
QP   5203 REM adresy 1536-1656 ($600-$678)
FI   5204 REM Przed wywolaniem trzeba
NK   5205 REM otwarty kanal do odczytu.
GA   5206 REM Wywolanie:
AJ   5207 REM I=USR(ADR(OKOAL$),kanal*16,K)
CK   5208 REM K=0 - brak odczytu kolorow
VQ   5209 REM K=1 - kolory sa odczytywane
AU   5210 REM Zwraca wartosc I rowna:
AN   5211 REM 1 gdy wszystko w porzadku
EQ   5212 REM lub numerowi bledu, ktory
YD   5213 REM wystapil przy zapisie.
XT   5214 REM Wartosc 255 oznacza probe
YN   5215 REM odczytu z pliku niezgodnego
QY   5216 REM z KOALA, a 254 - uzyta jest
YD   5217 REM inna kompresja niz pionowa.
JR   5219 REM
LD   5220 RESTORE 5300
OB   5225 FOR T=0 TO 5:S=0:FOR T1=1 TO 20:R
     EAD A:S=S+A:POKE 1535+T1+T*20,A:NEXT T
     1:READ A:IF S<>A THEN 5250
JX   5230 NEXT T
FO   5235 FOR T=6 TO 16:S=0:FOR T1=1 TO 20:
     READ A:S=S+A:OKOAL$(T1+(T-6)*20)=CHR$(
     A):NEXT T1:READ A:IF S<>A THEN 5250
KA   5240 NEXT T
BN   5245 RETURN
CD   5250 ? :? :? "Blad danych w wierszu ";
     T+5300:? :? :? :LIST 5300+T:STOP
MJ   5300 DATA 160,0,145,203,24,165,203,105
     ,80,133,203,144,2,230,204,198,207,208,
     47,169,2830
AA   5301 DATA 96,133,207,165,209,240,24,19
     8,209,230,205,208,2,230,206,165,205,13
     3,203,165,3433
YW   5302 DATA 206,133,204,198,208,206,19,1
     60,1,208,43,230,209,24,165,205,105,40,
     133,203,2902
BJ   5303 DATA 165,206,105,0,133,204,96,166
     ,214,169,0,157,72,3,157,73,3,169,7,157
     ,2256
FE   5304 DATA 66,3,32,86,228,48,1,96,192,1
     36,208,2,160,1,132,212,169,0,133,213,2
     118
AK   5305 DATA 166,215,154,96,1,0,26,199,20
     1,128,255,83,77,65,82,75,83,111,102,11
     6,2235
WB   5306 DATA 104,104,104,133,214,104,104,
     133,219,186,134,215,169,0,133,209,169,
     96,133,207,2870
PI   5307 DATA 169,40,133,208,165,88,133,20
     3,133,205,165,89,133,204,133,206,169,6
     ,133,218,2933
GM   5308 DATA 32,67,6,166,218,221,104,6,24
     0,5,160,255,76,94,6,198,218,16,237,32,
     2357
JE   5309 DATA 67,6,201,1,240,4,160,254,208
     ,236,160,10,165,219,240,2,160,5,132,21
     8,2690
XQ   5310 DATA 32,67,6,198,218,208,249,165,
     219,240,18,169,0,133,218,32,67,6,166,2
     18,2629
UQ   5311 DATA 157,196,2,230,218,224,4,208,
     242,169,9,133,218,32,67,6,198,218,208,
     249,2988
JA   5312 DATA 32,67,6,133,220,41,127,240,1
     0,170,202,134,216,169,0,133,217,240,17
     ,32,2406
BA   5313 DATA 67,6,56,233,1,133,216,8,32,6
     7,6,40,233,0,133,217,32,67,6,72,1625
DU   5314 DATA 104,72,32,0,6,165,220,16,5,1
     04,32,67,6,72,165,216,240,5,198,216,19
     41
UP   5315 DATA 24,144,233,5,217,240,6,198,2
     17,198,216,208,223,165,220,48,4,104,24
     ,144,2838
EV   5316 DATA 175,104,24,144,174,83,77,65,
     82,75,45,83,111,102,116,0,49,57,57,48,
     1671
QV   5317 REM ****
```

LISTING 4

```
NU   0 REM DEMONSTRACJA ODCZYTU KOALA
AC   1 REM Krzysztof Klimczak
DU   2 REM (c) 1991, Sp. Bajtek
NI   3 REM
PB   110 DIM OKOAL$(220):REM to musi byc
NZ   115 GOSUB 5200:REM inicjacja procedury
      maszynowej
UH   120 GRAPHICS 6+16:COLOR 1
SZ   125 OPEN #1,4,0,"D:PRZYKLAD.PIC":REM o
     twarcie "obrazka" do odczytu
UB   130 I=USR(ADR(OKOAL$),16,0)
XK   135 REM wywolanie procedury (16=1*16,
     0 - ominiecie aktualizacji kolorow, 1
     - aktualizacja)
LF   140 CLOSE #1
NB   145 IF I=255 THEN ? "TO NIE JEST ZBIOR
      KOALI":STOP
ZV   150 IF I=254 THEN ? "ZBIOR NIE JEST ZA
     PISANY W PIONOWEJ KOMPRESJI":STOP
AW   155 IF I>1 THEN ? "WYSTAPIL BLAD PRZY
     ODCZYCIE NR: ";I:STOP
WT   160 POKE 764,255
KK   165 IF PEEK(764)=255 THEN 165
OE   170 END
XX   175 REM ***
```

TOP 12

XL/XE

ZYBEX Zeppelin Games zręcznościowa	**C/D**	Po raz drugi na naszej liście gra, która wyróżnia się nie tylko bardzo dobrą grafiką, ale i stwarza warunki do świetnej zabawy także dla kilku osób. Strzelanina „na wyżycie się", ale zupełnie nietypowa, niesamowicie wciągająca, oczywiście ze znakomitą grafiką.
INFILATOR U.S. Gold przygodowa	**D**	Również po raz drugi. Tej gry nie powinno zabraknąć w żadnej „grotece". Dzięki rozbudowanej fabule atrakcyjna prawie dla każdego. Jeszcze raz przypomnienie: nie dajmy się nabrać na wersje kasetowe (tylko fragmenty programu).
BLINKY'S SCARY SCHOOL Zeppelin Games przygodowa	**C/D**	Po raz kolejny, ale trudno, aby na liście zabrakło tak starannie opracowanej gry. Grafika, animacja i szczegóły opracowane do perfekcji, wszystko uzupełnia niebanalna fabuła: sympatyczny duszek wędrujący po starym zamczysku, coś musi znaleźć, coś gdzieś przenieść.
THE GOONIES Activision przygodowa	**C/D**	Ponownie na liście, ale trudno się dziwić — Steven Spielberg nakręcił znakomity film, a Activision zrobiło na jego podstawie bardzo dobry program. Możliwość gry dwóch osób jednocześnie, kilka poziomów z zagadką do rozwiązania na każdym, bardzo sympatyczna grafika, a instrukcja raczej zbędna, może tylko popsuć zabawę naszym szarym komórkom.
SILENT SERVICE Microprose symulacyjna	**C/D**	Gra nie najnowsza, ale nadal jeden z najlepszych symulatorów. Jesteś dowódcą okrętu podwodnego i przed tobą szereg trudnych zadań. Bardzo łatwa obsługa i starannie opracowana grafika podnoszą i tak wysoką ocenę programu, którego zaletą jest spory realizm.
RIVER RAID Activisioln zręcznościowa	**C/D**	To chyba najbardziej klasyczna gra dla tego komputera, ale byłoby nietaktem nie umieszczenie tego już paroletniego hitu na naszej liście. Mały samolocik nad rzeką pełną niebezpieczeństw. Gra, która przyciągnie do komputera każdego, sympatyczna grafika i wysoka wartość „przydatności do grania"
BOULDER DASH First Star Software zręcznościowa	**C/D**	To druga z klasycznych gier na nasze „małe" Atari, w odróżnieniu od „River Raidu" wymaga poza zręcznością umiejętności szybkiego myślenia i kojarzenia. Sympatyczny Rockford znalazł się pod ziemią w otoczeniu kamieni i diamentów, te pierwsze musi omijać, te drugie zbierać. Wiele komnat, wiele rozwiązań i wciąż upływający czas.
HENRY'S HOUSE Mastertronic labiryntowa	**C/D**	Bardzo sympatyczny bohater Henry znalazł się w niesamowitym domu. Pełno tu latających przedmiotów, przeszkadzających urządzeń i niebezpiecznych pokoi, a trzeba je pokonać wszystkie — zadanie wbrew pozorom niełatwe. Miła dla ucha muzyka i bardzo dobra grafika sprawiają, że gra powinna się podobać wszystkim od lat 5 do 90.
FEUD Mastertronic przygodowa	**C/D**	Dawno, dawno temu była sobie wioska, mieszkało w niej dwóch braci czarowników, ale pewnego dnia pokłócili się i wypowiedzieli sobie wojnę. Jesteś jednym z nich, wędrujesz po wiosce zbierając zioła, które pozwolą ci rzucać zaklęcia potrzebne do pokonania przeciwnika. Świetna zabawa w bardzo dobrej grafice
EXPLOADING WALL Byte Back zręcznościowa	**C/D**	Zasada gry jest dobrze znana ze słynnego „Arkanoica" — odbijanie piłeczki w kierunku muru z cegiełek, uderzenia niszczą poszczególne elementy konstrukcji. Wykonanie natomiast jest znacznie lepsze — poruszamy się po czterech ekranach przewijanych podczas gry, staranne opracowanie graficzne, ciekawe efekty i wciągająca zabawa.
BATTLE SHIPS Mirage planszowa	**C/D**	Komputerowy przebój na „małe" Atari wykonany w Polsce, prawie nie różni się od swojego pierwowzoru (Atari ST). Znana wszystkim „gra w okręty", ale w ładnej oprawie, salwy burtowe, tonące statki, ataki samolotów, warianty dla jednego i dwóch graczy, bardzo dobra muzyka. Polecamy oczywiście oryginalnie wydany program.
ROBBO Avalon zręcznościowa	**C/D**	Kolejny polski akcent — gra już dość dobrze znana użytkownikom Atari, w założeniach trochę przypominająca „Boulder Dash". Mały robocik na obcej planecie musi pokonać kilkanaście plansz. Poprawna grafika i niezwykle wciągająca fabuła, niezwykle lubiana, doczekała się kilku przeróbek (pseudowersje Robbo II czy III).

Na zakończenie również parę słów o nowościach — wbrew panującym opiniom i plotkom jednak nadal pisze się programy na nasze „małe" Atari, a coraz więcej wartościowych programów zaczyna powstawać w Polsce.

RUFF AND REDDY (Hi-Tec) — **sympatyczna gra przygodowa, na podstawie komiksu Hanna-Barbera. Ruff i Reddy testują rakietę kieszonkową, tracą kontrolę i rozbijają się na obcej planecie, tam toczą boje z nieznanymi Lilli-Punies.**

HEAD OVER HEELS (Hit Squad) — **znana firma „Ocean" już dawno napisała tę popularną na innych komputerach grę, ale dopiero teraz zdecydowała się ją wydać i to firmując ją nie istniejącą Hit-Squad. Opowieść o sympatycznych stworkach wędrujących w trójwymiarowym labiryncie.**

TITAN (Atlantis Software) — **bardzo ładnie opracowana „strzelanina". Fabuła nie mająca większego znaczenia światu grozi zagłada, został już tylko jeden jedyny „wojownik", który może odwrócić bieg historii. Główną wartość gry stanowią dobra animacja i ładna grafika.**

ST

CADAVER Image Works przygodowa	Po raz drugi na naszej liście. Niezapomniana gra roku, znakomita grafika 3-D, rycerz wędrujący po labiryncie. Program, który otwiera nową epokę wśród gier przygodowych: pułapki, niezliczone przedmioty do zebrania, smoki, niesamowite stwory i niebezpieczni przeciwnicy. Świetna zabawa na długie godziny.
COLORADO Palace przygodowa	Również po raz drugi. „Colorado" należy do gier, które nie mogą się nie podobać, grafika, dźwięk, duża i dobrze animowana postać oczywiście bez zastrzeżeń. Nasz bohater jest traperem, który zapuścił się na dziewicze dla „białych" tereny, spotykani Indianie czasem bywają przyjaźni, ale tylko bywają...
KICK OFF II Anco sportowa	Trzeci z programów, który pozostał na naszej liście i chyba ma niewielkie szanse, aby ją opuścić, bo piłka nożna jest „chorobą narodową", a ta w wykonaniu firmy Anco bliska jest perfekcji. Nawet giełdowi piraci zapominają przy tej grze o swojej „profesji". Nie grafika, ale — jak mówią Anglicy „playability" decyduje o jej wartości.
PIPEMANIA Empire logiczna	Grafika — tak!, dźwięk — także!, ale przede wszystkim inne możliwości decydują o wartości tej zabawy. Jej zasada jest prosta: musimy w określonym czasie ułożyć jak najdłuższy rurociąg. Oczywiście można korzystać tylko z dostępnych elementów. Wydaje się to łatwe, ale już po kilku minutach gry okazuje się, jak trudno jest zrobić nawet małą przerwę, wystarczy spróbować i... noc z głowy.
MIDWINTER Rainbird przygodowa	Jest rok 2039, trwa zimna wojna między Wschodem i Zachodem. Na Florydzie zaobserwowano szybko zbliżający się niezidentyfikowany obiekt, który rozbija się we wschodniej Bhurmie, w promieniu 30 kilometrów wszystko zostaje zniszczone... Znakomita grafika, gra uważana za strategiczno-przygodową zabawę przyszłości, specjalna grupa rozpoczyna wyprawę ku nieznanemu. Oczywiście grafika i dźwięk równie wspaniałe jak cała gra.
PRINCE OF PERSIA Domark przygodowa	Gra nawiązuje do osławionego już „Karateki" — fabuła bardzo podobna: książę wędruje po niebezpiecznym zamku, gdzie spotyka świetnie wytrenowanych przeciwników, a dodatkowo sprytnie zamaskowane pułapki oraz wiele niespodzianek i tajemnic... Główną jej zaletę stanowi znakomita animacja, a staranne opracowanie graficzne uzupełniają bardzo realistyczne efekty dźwiękowe.
TENNIS CUP U.S. GOLD sportowa	Tytuł mówi wszystko — znakomita symulacja gry w tenisa: ekran podzielony na pół pozwala obserwować grę zarówno tak, jak widzi to jeden, jak i drugi gracz, głos sędziego informuje o zdobytych i straconych punktach, a dobywające się z głośnika odgłosy odbijanej piłki sprawiają wrażenie, jakbyśmy naprawdę siedzieli na korcie.
SKIDZ Gremlin zręcznościowa	Masz okazję spróbować swego szczęścia w dwóch dyscyplinach będących domeną dzisiejszej młodzieży: BMX i deskorolka. Ładna grafika i wiele atrakcyjnych wariantów: sklep, w którym można kupić wyposażenie, różne trasy, różne stopnie trudności, wiele niespodzianek na trasie. W sumie przyzwoita zabawa sportowa bez wstawania z krzesła.
F-19 STEALTH FIGHTER Microprose symulacyjna	Program, który angielskie czasopismo ST Action oceniło jako bliski ideału. Grafika bez zarzutu (obiekty quasi 3-D), symulacja bliska rzeczywistości, ponad 4000 misji do wykonania, niesamowite walki i pojedynki, bardzo dużo wariantów, mapy, klasyfikacje. Spędzając kilka godzin z tym programem można zostać znakomitym pilotem.
LOST PATROL Ocean zręcznościowa	Jest to gra strategiczno-zręcznościowa. 7 czerwca 1966 roku helikopter U.S. Army rozbija się w centrum Sajgonu, a ty zostajesz dowódcą siedmioosobowej grupy, która ma do pokonania 57 mil do Do Hoc, najbliższej amerykańskiej bazy. Digitalizowane obrazy, świetnie rozwiązane sceny walki, dźwięk karabinów maszynowych, wybuchy granatów — wszystko bliskie ideału.
POWERMONGER Electronic Arts ekonomiczna	Gra w swoich założeniach przypominająca osławionego „Populousa". Walce światów i przeciwników towarzyszy znakomita grafika. Gra, która pozwala nam sprawdzić samego siebie — wymaga niewątpliwie logicznego myślenia, umiejętności analizowania i cierpliwości. Końcowe zwycięstwo niewątpliwie sprawi nam dużą satysfakcję.
CRYSTALS OF ARBOREA Palace role-playing	Program, który w przeciwieństwie do innych tego rodzaju wyróżnia się fenomenalną grafiką. Świat został opanowany przez złego Morgotha — Lorda Chaosu, który sieje zniszczenie i śmierć. Jedyną nadzieją dla świata jest odnalezienie na maleńkiej wysepce Arborea czterech kryształów, próbuje tego dokonać książę Jarel wraz z drużyną.

Teraz kolej na trzy nowości, których będziemy niecierpliwie oczekiwali, a jak pokazuje rzeczywistość, nowości z poprzedniego odcnika już pokazały się na naszych „giełdach":

WRATH OF THE DEMON (Readysoft/Empire) — **gra przygodowa o bajecznej grafice, świetna animacja. Spokojne królestwo zostało zaatakowane przez zło — monstra, diabły, smoki, okropne stwory to twoi przeciwnicy podczas wędrówki.**

TIME MACHINE (Activision) — **jest to niezbyt klasyczna gra fantasy role-playing, ciekawostką jest podróż w czwartym wymiarze przy użyciu maszyny stworzonej przez profesora Potts'a, zaś firma jest gwarancją jakości.**

BADLANDS PETE (ARC) — **prawdziwy Dziki Zachód ze wszystkimi swoimi „specjalnościami": zniknęła córka gubernatora i wezwano na pomoc szybkostrzelnego Pete Coyote'a. Świetna, ale zarazem dowcipna grafika i oczywiście dźwięk bez zarzutu.**

FORTRESS

XL/XE

Jeśli przyjrzec się bliżej wielu popularnym grom to okazuje się, iż zwykle są to programy o stosunkowo prostej grafice i niewyszukanym dźwięku. Najlepszym przykładem może być „River Raid", który cieszy się niesłabnącym powodzeniem wśród wielu graczy, pomimo że ma już wiele lat.

Za jego odpowiednik w grupie gier strategicznych można z pewnością uznać program „Fortress" napisany w 1983 roku w firmie Iconographics Inc. Jego bardzo proste zasady sprawiają, że potrafi zainteresować niemal każdego — wielokrotnie z przyjemnością obserwowałem nawet dziesięcioletnie dzieci z pasją budujące zamki. Zresztą zwykle bardzo skutecznie — najsłabszy przeciwnik komputerowy przeważnie z nimi przegrywa.

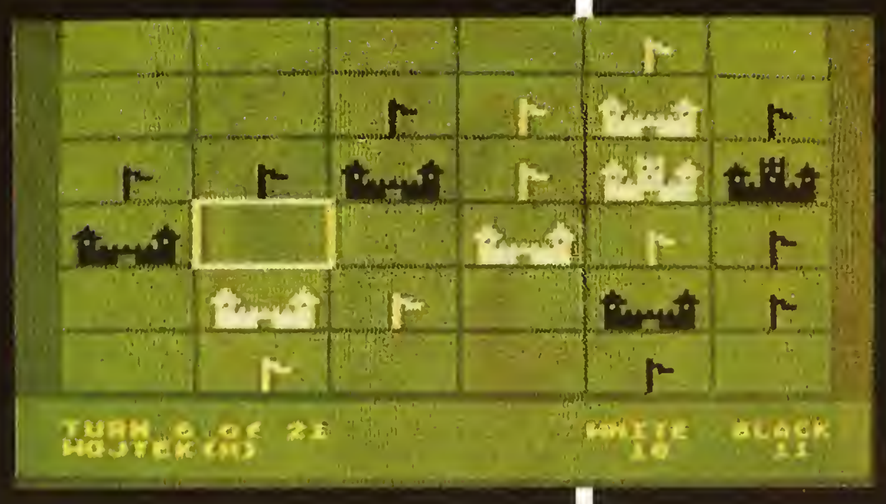
Coż takiego interesującego jest w tej grze? Polega ona na umieszczaniu fortec (stąd nazwa) na planszy o rozmiarach 6 na 6 pól. Kazda forteca obejmuje swoim wpływem cztery sasiednie pola, co jest sygnalizowane przez pojawienie się na nich chorągiewek. Wpływy fortec tego samego gracza dodają się, więc wzmacniają się one nawzajem. Natomiast wpływy fortec przeciwników się znoszą i w ten sposób powstają pola „ziemi niczyjej". Jeżeli wpływ sąsiadujących fortec przeciwnika jest zbyt duży, najpierw zamykana jest brama fortecy, a gdy ten wpływ jeszcze wzrośnie, to forteca... znika. Można temu zapobiec wzmacniając ją przez postawienie obok innych swoich fortec lub przez bezpośrednie poprawienie jej siły. W tym drugim przypadku najpierw buduje się w twierdzy wieżę, a potem dodaje się do niej jeszcze daszek. W taki sposób plansza zapełnia się białymi i czarnymi fortecami — wygrywa ten z graczy, który po upływie określonej liczby rund (normalnie 21) zajmuje więcej pól.

Te proste zasady są urozmaicone wieloma dodatkowymi możliwościami programu. Przede wszystkim można grać zarówno z komputerem, jak i z kolegą. Na dyskietce z programem zapisanych jest kilku graczy komputerowych o różnym stopniu umiejętnosci. Gracze ci stosują również rożną taktykę walki — od wybitnie defensywnej COUNT VAUBANa do bardzo agresywnej prezentowanej przez GENGIS KHANa. Istnieje ponadto możliwość samodzielnego zaprojektowania przeciwnika. To jednak jeszcze nie wszystko. Niewątpliwą atrakcją jest uczenie się przeciwnika. Tak, tak, to nie pomyłka — grając kilka kolejnych rund z jednym przeciwnikiem łatwo można zauważyć, że gra on coraz lepiej doskonaląc swoją taktykę. Nabyte przez komputerowego gracza umiejętności można zapisać na dyskietce i w kolejnej rozgrywce będzie on już dysponował poprzednim doświadczeniem stając się z czasem coraz groźniejszym przeciwnikiem. Dodatkową atrakcją jest możliwość zapisania na dyskietce także całej historii i punktacji stoczonych walk.

Wszystkie te atrakcje uzyskano w Basicu (!) przy użyciu naprawdę skromnych środków. Plansza ma kolor zielony, fortece są białe i czarne, trudno więc mówić o wspaniałej grafice. Efekty dźwiękowe ograniczają się natomiast do odgłosu podobnego do gongu przy drzwiach, gdy znika ktoraś z fortec. Mimo tej prostoty można spedzic na zabawie nawet cały dzień — niekiedy również noc!

(ziew)

GRY



Producenci: Iconographics
Strategic Simulation Inc.
Autorzy: Jim Templeman
i Patty Denbrook

Ocena w skali 1–10:	
GRAFIKA	4
DŹWIĘK	1
SENS GRY	8

DECISION IN DESERT

XL/XE

Gra „Crusade in Europe", której opis był zamieszczony w „Moim Atari 1", a krótka instrukcja znajduje się w tym numerze, stanowi pierwszą z trzech części cyklu gier strategicznych firmy Microprose. Drugim programem z tekj serii jest „Decision in Desert", którego tematem są walki w Afryce Północno-Wschodniej znane w historii pod wspólną nazwą kampanii libijskich.

Zarówno nazwa firmy, jak i nazwiska autorów stanowią wystarczającą zachętę do zagrania. Wszystkich, ktorzy znają już „Crusade in Europe" nie trzeba z pewnością namawiać do zainteresowania się tą grą. Po sukcesach we Francji, Belgii i Holandii warto spróbować swych sił na pustynnych piaskach. Zawarte w grze scenariusze obejmują bitwę pod Sidi Barrani, wyzwolenie Tobruku (operacja „Crusader") oraz bitwy pod El-Gazala, El-Alamein i Alam Halfa (nazewnictwo dwóch ostatnich jest w polskiej historiografii odwrotne). Oczywiście każdy ze scenariuszy ma co najmniej dwa warianty. Razem scenariusze te obejmują całość zmagań w Afryce Północno-Wschodniej, aż do lądowania Aliantów w Afryce Północno-Zachodniej.

Gra wiernie odzwierciedla realia pola walki, podobnie jak w „Crusade in Europe". Ponieważ jednak część czytelników nie miała okazji zapoznać się z jej opisem, to przypomnijmy najważniejsze elementy. Na ekranie widać tylko część jednostek przeciwnika — są to te, które znajdują się w pobliżu własnych oddziałów lub zostały wykryte przez rozpoznanie lotnicze. Wartość bojowa każdej jednostki jest uzależniona od jej doświadczenia, stopnia zużycia sprzętu i zmęczenia żołnierzy oraz od poziomu zaopatrzenia. Na skuteczność działań i ruchliwość oddziału wpływa także formacja, w jakiej się on znajduje. Na przykład, jednostka ufortyfikowana jest bardzo silna w obronie, lecz jej przesunięcie zajmuje wiele czasu, koniecznego do przejścia w szyk rozproszony. Jednostki przemieszczane na duże odległości przyjmują szyk marszowy lub transportowy, w którym wszakże z trudnością się bronią i słabo atakują.

Najważniejszą cechą odróżniającą programy z tej serii od innych gier strategicznych jest ich płynność. Brak tu bowiem podziału na fazy (ruchu, strzelania itd.), a walka toczy się ciągle. Wszystkie oddziały przejawiają pewną samodzielność i niekiedy zmieniają swoje postępowanie w zależności od sytuacji, nie bacząc na wydane rozkazy. Na przykład, atakująca jednostka po napotkaniu zbyt dużego oporu nieprzyjaciela samodzielnie przechodzi do obrony.

Zachęcam wszystkich miłośników gier strategicznych do wypróbowania swych sił w pustynnych walkach, a z pewnością nie będą zawiedzeni. Gra ta stanowi niewątpliwie kolejny liść w laurowym wieńcu firmy MicroProse.

(ziew)

Producent: MicroProse Software
Autorzy: Sid Meier i Ed Bever

Ocena w skali 1–10:	
GRAFIKA	8
DŹWIĘK	5
REALNOŚĆ	9
SENS GRY	10

CRYSTALS OF ARBOREA

ST

Nadeszły ponure czasy przynosząc nędzę, biedę, choroby, ubóstwo i nieszczęścia, świat został opanowany przez złego Morgoth'a — Lorda Chaosu, pana śmierci i zniszczenia. W tych niesamowitych czasach Bogowie umieścili losy świata na niewielkiej wysepce o nazwie Arborea, gdzie Morgoth ukrył skradzione kryształy harmonii. Jednak na „arenie" pojawił się wspaniałomyślny książę Jarel, który wyruszył w bój z Lordem Chaosu. Książę musi najpierw skompletować drużynę, ma do wyboru sześciu wspaniałych wojowników, każdy z nich ma inne wady i zalety, Jon, Zach, Irvan, Akeer, Olbar, Thorm. Każdy z członków drużyny posiada określone cechy fizyczne (jak energia, ruchliwość, witalność, siła, itp.) oraz może używać określonych zaklęć i czarów.

Drużyna podczas wędrówki będzie spotykać różnych niesamowitych przeciwników: gnomy, orki, diabły, itp. Z wyspy nie ma ucieczki, więc należy podjąć walkę — raz skuteczny może być atak wojownika, czy rycerza, innym razem zaklęcia czarownika. Wędrówka, trwa dość długo, więc drużynę może zastać także noc, jednak książę Jarel ma specjalną zdolność widzenia w nocy (Night Vision) i tylko on może trzymać w ręku tajemnicze kryształy, których miejsce jest w świątyni...

Firma „Palace" nie zwykła robić gier złych lub przeciętnych, „Crystals of Arborea" potrzymują tę tradycję. Znakomita grafika to rzadkość wśród gier role-playing, a zarazem jest ona jednym z większych atutów tego programu. Niezliczona ilość miejsc, przedmiotów, sytuacji, przeciwników i dobry dźwięk jeszcze bardziej podnoszą ocenę ogólną. Zupełnie inaczej rozwiązano sceny walki, wprowadzając znacznie większe możliwości kierowania strategią walki. Jest to na pewno jedna z najlepszych gier typu role-playing i powinna się znaleźć w każdej „grotece" zaawansowanego gracza.

(tom)

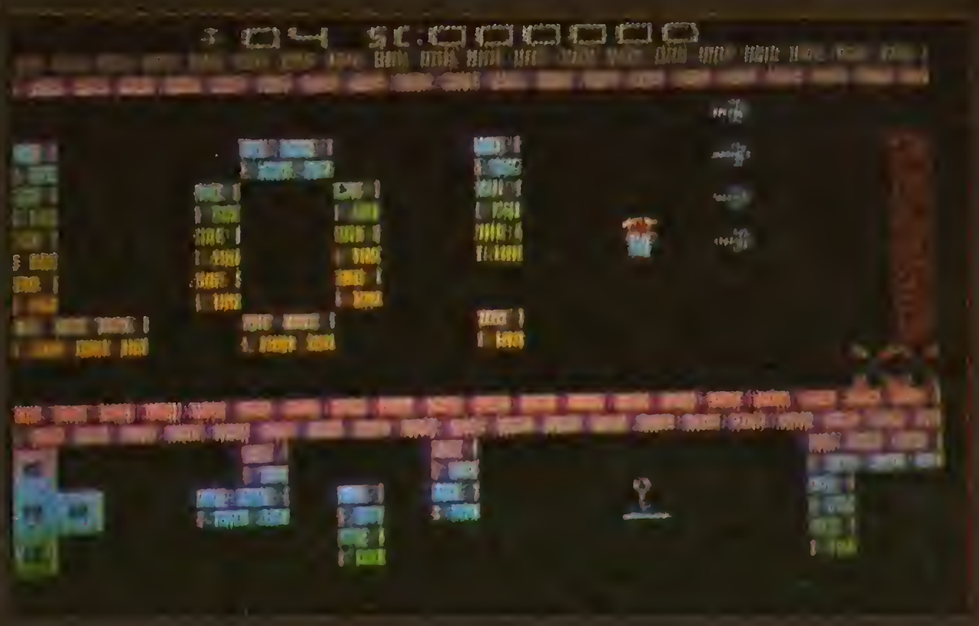
Producent: Palace
Autorzy: Pascal Einsweiler i Michel Perrot
Muzyka: Fabrice Hautecloque

Ocena w skali 1–10:	
GRAFIKA	8
DŹWIĘK	9
SENS GRY	9

GRY

SARACEN

XL/XE



To jednak miłe, iż komuś zdarza się jeszcze czasem napisać grę nie wymagającą od użytkownika instynktu mordercy i w dodatku reprezentującą sobą jakikolwiek poziom. Jeżeli autor umieści akcję w zawsze modnym labiryncie, obwaruje zwycięstwo koniecznością myślenia i ponadto pobudzi ambicję graczy niezliczoną ilością poziomów, pozostaje mu tylko nadać produktowi tytuł Saracen, żebyśmy wszyscy wiedzieli o co chodzi.

Mimo najlepszych chęci twórcy treść gry jest nieco monotonna. Niewielki ludzik, sądząc z wyglądu blondynek, ma za zadanie pokonywanie kolejnych poziomów i poważnych ilości przeszkód, bez żadnego konkretnego powodu bądź motywacji. Aby osiągnąć swój cel, musi w określony sposób wykorzystać odpowiednie przedmioty, których jest niestety niewiele. Brak jest ograniczenia czasowego — tempo zabawy zależy w 100% od gracza. Grafika nie sprawia raczej złudzenia oglądania filmu animowanego, aczkolwiek jest staranna i czytelna. Na efekty dźwiękowe składa się kilka taktów melodii, nie „rzucającej się w uszy", niemniej jednak zawsze możliwej do wyłączenia przy użyciu potencjometru głosu w monitorze.

Nie zdradzam, jak przejść I planszę Saracena, gdyż po poznaniu zastosowanych w niej tricków nie należy oczekiwać w tym zakresie niczego nowego. Nie oznacza to wcale, że Saracen jest grą po prostu nudną i niewartą zainteresowania. Tajemnicza liczba poziomów intryguje w najwyższym stopniu, nie pozwalając przerwać gry. Chyba, że ktoś jest mało ambitny. Ale wtedy zawsze może po prostu sformatować dyskietkę.

(Master)

Firma: Intelicreations
Autor: Ilan Ginsburg

Ocena w skali 1–10:	
GRAFIKA	5
DŹWIĘK	5
SENS GRY	6

LOST PATROL

ST

7 czerwca 1966 roku, helikopter U.S. Army z siódemką doborowych amerykańskich żołnierzy rozbija się w samym centrum Sajgonu. Aby uratować życie, oddział musi przejść 57 mil. Droga pełna jest pułapek, niebezpiecznych miejsc i zdecydowanych na wszystko żołnierzy Vietcongu, ale jedyną szansą jest jak najszybsze osiągnięcie celu, czyli amerykańskiej bazy w Do Hoc. Doświadczony dowódca (dwa lata w czynnej służbie) sierżant Charlie Weaver wie, że jest to żądanie bardzo trudne, posiadają niewielkie ilości żywności i amunicji, ale nadchodzi noc i to może okazać się ich największą szansą, pozostaje tylko nadzieja...

Na szczęście oddział nie składa się z nowicjuszy:

William Blom — 24 lata, sześć miesięcy w Wietnamie, drugi dan Aikido, czarny pas;

Robert Case — 31 lat, dziewięć miesięcy w Wietnamie;

Harvey Moore — 22 lata, sześć miesięcy w Wietnamie;

Richard Bachman — 28 lat, 14 miesięcy w wojsku, sześć miesięcy w Wietnamie, znakomity strzelec wyborowy;

David Cain 27 lat, dwa lata w wojsku, siedem miesięcy w Wietnamie;

Juan Gomez — 24 lata, dziewięć miesięcy w Wietnamie.

Natychmiast po katastrofie spróbowali pozbierać amunicję i żywność, ale nie uzbierało się tego dużo, dlatego będą zmuszeni „złożyć wizytę" w pobliskiej wiosce. Przodem został wysłany zwiadowca Gomez, obrano kierunek na północ. Jednak na drodze znalazł się nieprzyjacielski bunkier, bardzo by się przydał wyborowy snajper, ale Bachman jest ranny w lewe ramię. Atak może okazać się czynem zdecydowanie desperackim, a morale w grupie jest bardzo niskie...

Ty, jako gracz kierujesz oddziałem rozbitków. Zadanie jest proste, aby uratować własną skórę musisz jak najszybciej dotrzeć do Do Hoc. „Lost Patrol" jest jedną z najlepszych strategicznych gier akcji.

Znakomita grafika, szczególna dbałość nawet o najdrobniejsze szczegóły, często pojawiają się digitalizowane obrazy, elemetny powiększeń i zbliżeń dopracowane do perfekcji. Dźwięk należy również do atutów programu, salwy karabinów maszynowych, eksplozje granatów są tak realistyczne jakbyśmy znaleźli się w sercu bitwy. Kolejną zaletą programu jest znakomite połączenie elementów gry strategicznej z klasyczną grą typu „arcade", sceny walki, pojedynki snajperskie niezwykle uatrakcyjniają całość.

„Lost Patrol" nie jest grą dla „łamaczy joysticków", a godną polecenia zabawą dla tych, którzy chcą trochę pomyśleć, nie zapominając jednak o umiejętnościach manualnych, dla tych, którzy chcą przeżyć prawdziwą męską przygodę nie wstając z fotela i nie narażając się na rzeczywiste niebezpieczeństwa. Dodatkowo firma „Ocean" jest gwarantem naprawdę wysokiej jakości.

(tom)

Producent: Ocean
Autor: Simon Cooxe

Ocena w skali 1–10:	
GRAFIKA	10
DŹWIĘK	8
SENS GRY	7

TAIPEI

XL/XE

Podobnie jak opisywana wcześniej „Fortress" również i ta gra nie epatuje wspaniałymi efektami dźwiękowymi i graficznymi, lecz potrafi wciągnąć do zabawy. Także i tu idea jest tak prosta, jak tylko można to sobie wyobrazić, a efekt zaskakująco dobry.

Mówiąc najprościej, jest to zabawa klockami. Klocki te zostają ułożone w piramidkę i trzeba je po prostu zebrać. Kłopot polega na tym, że zbieranie podlega pewnym zasadom (równie prostym jak cała idea gry). Wszystkie klocki, których jest 144, są oznakowane różnymi symbolami. Każdy symbol powtarza się na czterech klockach. Klocki należy zbierać parami — po dwa identycznie oznakowane i pod warunkiem, że oba nie są zablokowane przez inne klocki.

Gra jest sterowana joystickiem, który służy do wskazywania zbieranych klocków. Dodatkowe funkcje gry wybiera się z rozwijanych menu umieszczonych u góry ekranu (jak w ST). Menu te umożliwiają wybór czasu trwania gry (5, 10 lub 15 minut albo bez ograniczenia czasowego), powtórzenie od początku ostatniej gry, grę demonstracyjną, wskazanie klocków, które można zebrać (jeśli gracz sam ich nie może znaleźć) oraz cofnięcie kilku ostatnio wykonanych ruchów. Aktualny stan gry można odczytać ze znajdującego się w dolnej części ekranu wiersza, w którym podana jest liczba pozostałych klocków i posunięć do wykonania oraz upływający czas.

Wszystko to jest prezentowane w ubogiej szacie graficznej — rysunki są czarno-białe, a jedynym urozmaiceniem są pasy w dolnej i górnej części ekranu oraz użyta w nich charakterystyczna czcionka. Jedynym efektem dźwiękowym jest natomiast obrzydliwy warkot, wydobywający się z głośnika przy błędnym wskazaniu klocków do zebrania. Skromność oprawy jest jednak zaletą tego programu, nie powoduje bowiem rozpraszania gracza na mało istotnych szczegółach i pozwala skupić się na rozgrywce.

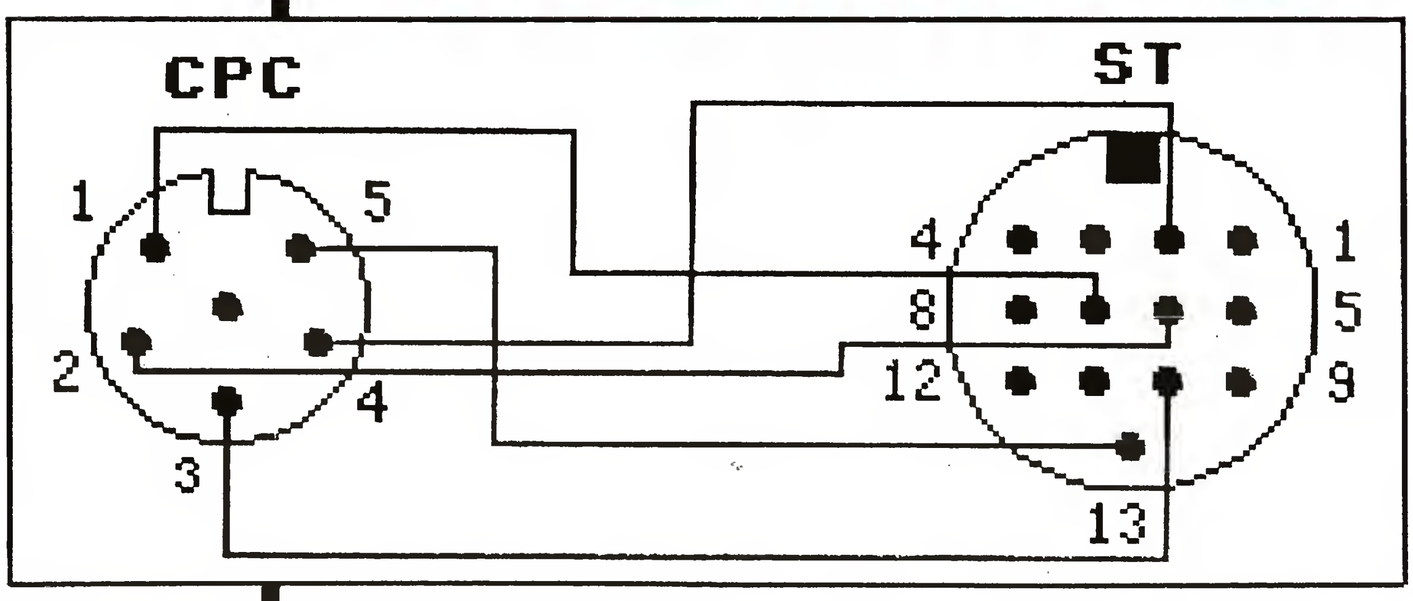
A jest się na czym skupiać. Prostota gry jest tylko pozorna, gdyż często zdarzają się klocki tak położone, że można je zebrać wyłącznie w określonej kolejności. Czasem wychodzi to na jaw dopiero pod koniec gry i trzeba się sporo cofnąć, aby poprawić popełniony błąd. Niekiedy w takim wypadku nie pozostaje nic innego, jak tylko zacząć od początku. Radzę spróbować.

(ziew)

Autor: Martin Lange

Ocena w skali 1–10:	
GRAFIKA	5
DŹWIĘK	1
SENS GRY	8

ST Z MONITOREM AMSTRADA



Do Atari ST można przyłączyć monitor monochromatyczny lub kolorowy. Często zamiast monitora kolorowego wykorzystywany jest telewizor. Jest to jednak rozwiązanie kłopotliwe i niewygodne.

Niekiedy można zakupić tanio kolorowy monitor przeznaczony do komputera Amstrad/Schneider CPC. Zdarza się również, że monitor CTM 640 pozostaje po zamianie komputera CPC na Atari ST. Monitor ten może współpracować z ST — trzeba jedynie wykonać odpowiedni przewód połączeniowy. W tabelkach zamieszczonych obok podane są sygnały dostępne w gnieździe komputera ST i monitora CTM, na rysunku zaś pokazany jest schemat połączeń przewodu.

Marek ZACHAR

Złącze monitorowe Atari

styk	sygnał
1	Audio Out
2	Composite Sync
3	General Out
4	Monochrom Sensor
5	Audio In
6	Green
7	Red
8	+12 V
9	Horizontal Sync
10	Blue
11	Monochrom Signal
12	Vertical Sync
13	Ground

Złącze monitorowe Amstrad

styk	sygnał
1	Red
2	Green
3	Blue
4	Composite Sync
5	Ground
6	Luminance

ATARI STUDIO "AS"

Korespondencja: "AS" Warszawa 33 P.O.Box 80

Firma: ul.Gen.Abrahama 4 (dawna Jóźwiaka) tel.125-123

ATARI STUDIO i MIRAGE Software polecają Państwu oraz firmom komputerowym oryginalne polskie programy na ATARI

BATTLE SHIPS - znana z innych komputerów gra w okręty, dla 1 lub 2
ORTOGRAFIA - świetna zabwa edukacyjna, rywalizacja, ładna grafika
GEOGRAFIA POLSKI - encyklopedia i test z geografii gospodarczej
RODERIC - w starym zamku, gra przygodowa w stylu Spellbound'a
BOMB JANE BAD - gra zręcznościowa przypominająca Bomb Jack'a
BERTYX - gra labiryntowa, wędrówka po nieznanej planecie
CO MOŻNA ZROBIĆ Z KOMPUTEREM ATARI - książka + kaseta/dysk

ZAPRASZAMY DO WSPÓŁPRACY WSZYSTKICH, KTÓRZY PISZĄ LUB CHCIELIBY PISAĆ PROGRAMY, NIE TYLKO NA ATARI !

ZMIENILIŚMY ADRES

Redakcja Mojego Atari
mieści się teraz
przy ul. Wasilkowskiego 7
tel. 643-18-40

Atari Turbo 2000 F

Nowy system transmisji danych z magnetofonem przyspieszony do 6700 bodów.

Komplet:
— cartridge
— oprogramowanie
— przeróbka magnetofonu
— instrukcja obsługi
— 12 miesięcy gwarancji

Instalacje wykonujemy na poczekaniu.

Interfejs do zwykłego magnetofonu.

Duży wybór oprogramowania w standardzie TURBO-2000.

Informacja: **Tel. 33-40-91**

Korespondencja:

MUEL

Ul.Cząstkowska 30
01-678 Warszawa

B82

REKLAMUJ SIĘ W MOIM ATARI

BIURO REKLAM CZYNNE CODZIENNIE W GODZ. 9.00–15.00

Najtańsze w Polsce!

SV 119
Junior
2 Fire
6 Blaszanych styków
Prosty mechanizm

SV 120
Junior–Stick
2 Fire
6 Blaszanych styków
Uchwyt pistoletowy

SV 122
Quickjoy II
2 Fire
6 Blaszanych styków
AutoFire
Drążek lotniczy

SV 124
Turbo
6 Mikrostyków
AutoFire
Drążek lotniczy

SV 123
Supercharger
2 Fire
6 Mikrostyków
Ergonomiczna budowa
Precyzyjny mechanizm

SV 126
Jet Fighter
2 Fire
6 Mikrostyków
AutoFire
ACS-Regulator
szybkości AUTO
Obsługa pod kciuk"
Drążek lotniczy

SV 125
Superboard
6 Fire
10 Mikrostyków
AutoFire
Cyfrowy wyświetlacz
czasu
Sygnał dźwiękowy
Przełącznik dla
leworęcznych
Drążek lotniczy

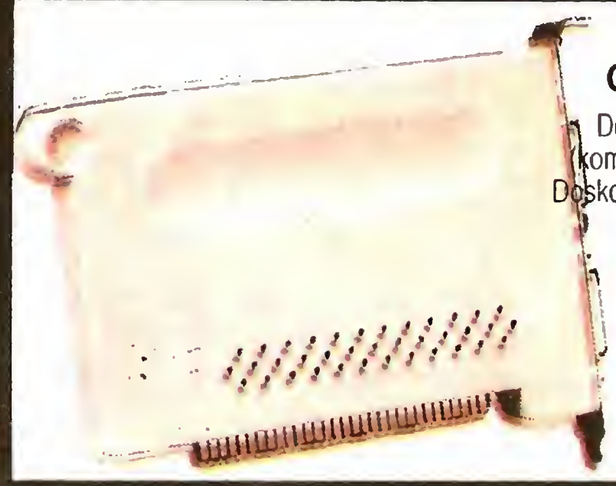
SV 130
IR Infrared
1 Fire
5 Mikrostyków
Podczerwień
Daleki zasięg
Odbiornik

SV 128
Megaboard
4 Fire
10 Mikrostyków
AutoFire
6 cyfrowy stoper
ATM — Anti Tilt Mechanism
Fire Pad

SV 140
Enterprice
2 Fire
6 Mikrostyków
AutoFire
ACS — Regulator
szybkości AUTO
Drążek lotniczy
„kierownica"
Kabel 4 m

SV 201
Quickjoy M 5
Do IBM XT/AT
(kompatybilnych)
Współpracuje z Game-Card
lub I/O Card
2 Fire
2 AutoFire
6 Mikrostyków
Wybór AUTO
PSC — Regulator XY
Sygnalizacja Świetlna
Fire
ASC — Regulator szybkości
AUTO

SV 202
M 6 analog
Analogowy
DO IBM XT/AT
(kompatybilnych)
Współpracuje z Game-Card
lub I/O Card
2 Fire



SV 210
Game Card
Do IBM XT/AT
(kompatybilnych)
Doskonale pracuje
z M 5 i M 6

SV 127
Top Star
2 Fire
6 Mikrostyków
AutoFire
Przeźroczysta obudowa
SAS — Shock Absorbing
System
Platynowane części

SV 500
Van 3
Pudełka na dyskietki
80 sztuk 3½"
Zamknięcie na klucz

SV 510
Van 5
Pudełko na dyskietki
80 sztuk 5¼"
Zamknięcie na klucz

SV 119 — 64 900	SV 123 — 119 900	SV 128 — 319 000	SV 210 — 219 000
SV 120 — 74 900	SV 125 — 239 900	SV 130 — 339 000	SV 500 — 119 000
SV 122 — 89 900	SV 126 — 169 900	SV 201 — 249 000	SV 510 — 119 000
SV 124 — 109 900	SV 127 — 249 900	SV 202 — 229 000	

SV 201 + SV 210 — 439 000
SV 202 + SV 210 — 399 000

interwencje
tel. 23-92-21

Należy przesłać pieniądze przekazem zwykłym (czerwony blankiet) na adres:

TAL Quickjoy

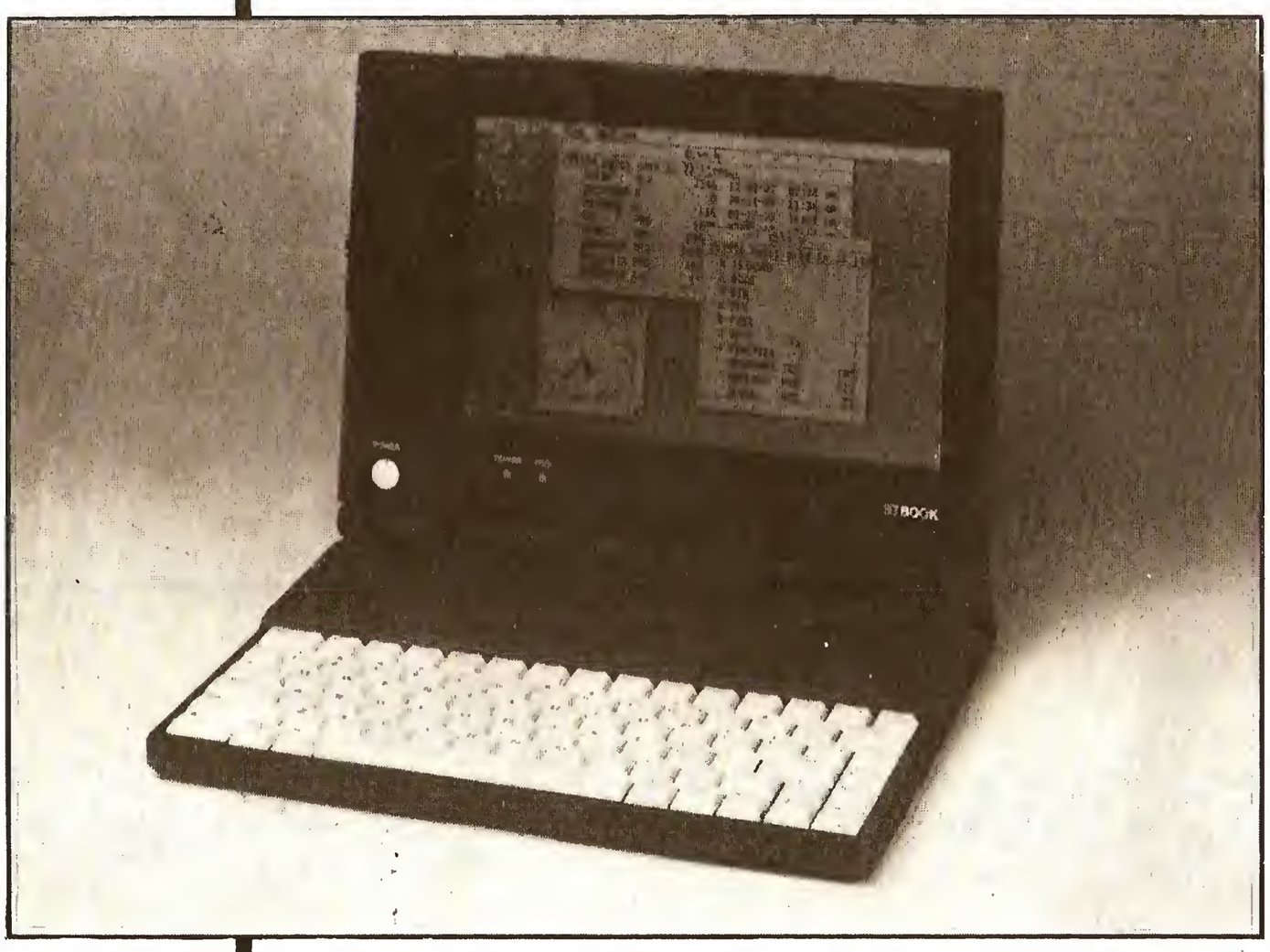
***Uwaga!** W miejscu na korespondencję prosimy podać po raz drugi: adres domowy, kod i symbol zamówionej pozycji. W przypadku braku adresu zwrotnego nie realizujemy zamówienia.*

„TAL" — QWERTY
ul. Mikowa 45
02-411 Warszawa
fax: 659-12-35

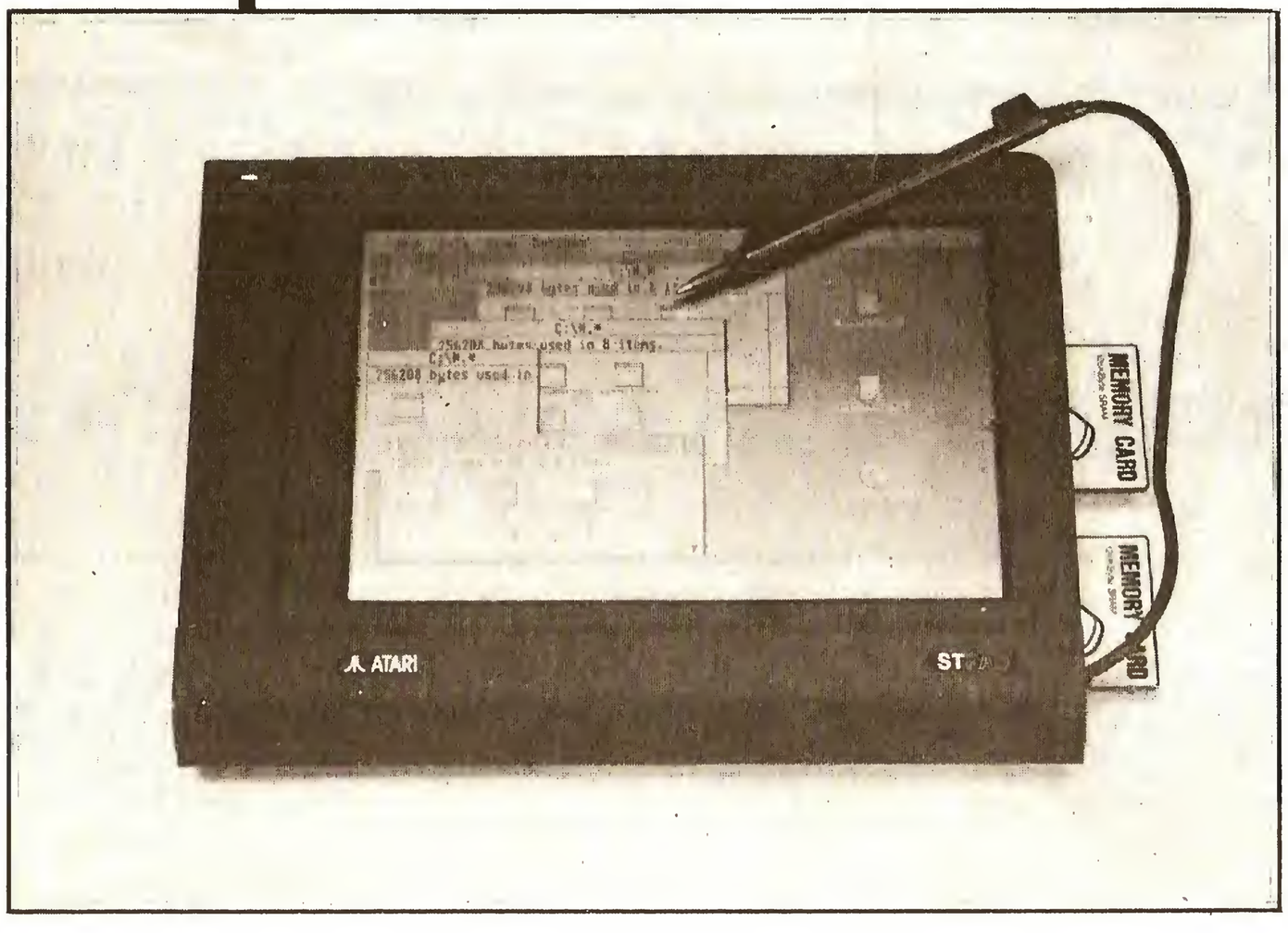
Czas realizacji 14 dni

ATA na CeBIT-cie

Hasło przewodnie tegorocznej ekspozycji Atari na CeBIT-cie brzmiało „Komputer bliżej człowieka". Na wystawie w Hanoverze firma zaprezentowała interesujące prototypy urządzeń, które już w II połowie tego roku mają znaleźć się w sieci specjalistycznych sklepów. Produkty te są efektem intensywnych prac nad dalszą miniaturyzacją, zwiększeniem mocy obliczeniowej i ułatwieniem obsługi.



Elektroniczny notes STBook z twardym dyskiem 20 MB i 1 MB pamięci roboczej.



Po ekranie STPad można pisać specjalnym piórem jak na zwykłej kartce papieru.

Najnowsze modele Atari można bez problemu nosić w teczce. Przełamano kolejną barierę dzielącą nieufnego i niedoświadczonego użytkownika od komputera. Poznanie zasad prostego języka, w którym należy porozumiewać się z elektronicznym urządzeniem, nie wymaga ukończenia studiów informatycznych. Praca z nowoczesnym komputerem przypomina raczej czytanie ciekawego, kolorowego komiksu, po który może sięgnąć każdy, niezależnie od wieku i wykształcenia.

STBook

Szczególnie silne wrazenie na zwiedzających targi wywierał maleńki, przenośny STBook. Jest on reprezentantem rodziny komputerów typu notebook, która od niedawna zyskała ogromną popularność wśród użytkowników. Notebook, czyli po angielsku notes, ma podstawowe cechy konstrukcyjne zbliżone do laptopa, lecz jest znacznie lżejszy i mniejszy. Moze on spełniać rolę elektronicznego notesu, który zastąpi tradycyjny, papierowy brulion. Dzięki małym wymiarom i niewielkiej wadze przenoszenie STBooka nie sprawia żadnych kłopotów. Komputer ten może towarzyszyć nam w każdej podróży i oddawać nieocenione usługi w bardzo wielu sytuacjach.

STBook waży zaledwie około jednego kilograma i ma wymiary 30 x 21 cm. To maleństwo kryje w sobie kompletny komputer ST z twardym dyskiem 20 MB i 1 lub 4 MB pamięci RAM. Niestety miniaturyzacja ma też swoją cenę. W tym przenośnym urządzeniu zabrakło już miejsca na stację dysków elastycznych. Klawiatura składająca się z 84 (lub 85) klawiszy odpowiada klawiaturze ST i TT.

Laptopy i notebooki są niejednokrotnie eksploatowane w warunkach polowych. Zwykle trzyma się je. podczas sporządzania notatek, po prostu na kolanach. W takich warunkach zastosowanie myszy jest niemożliwe. Dla użytkowników przyzwyczajonych od posługiwania się myszą skonstruowano specjalny, wbudowany na stałe manipulator, który nosi nazwę Joypad.

Ekran ciekłokrystaliczny jest umieszczony w składanej pokrywie komputera. Rozdzielczość obrazu wynosi 640 x 400 punktów. Atari wyposażyła swoje najnowsze dziecko we wszystkie najniezbędniejsze złącza. Ma on interfejs równoległy Centronics, interfejs szeregowy RS 232, 2 gniazda MIDI, złącze DMA, gniazdo zewnętrznej stacji dysków i złącze komunikacyjne.

Do komputera można dokupić dodatkowe wyposażenie, które znacznie poszerza jego i tak spore możliwości. Użytkownik STBooka może zażyczyć sobie zainstalowanie karty modemu telefaxu, twardego dysku o pojemności 40 lub 60 MB, klawiaturę numeryczną lub mysz. Przewiduje się także dołączenie do komputera zewnętrznej stacji dysków elastycznych o pojemności 1,44 MB.

STBook jest zasilany z akumulatorków, które ładuje się przy pomocy specjalnego zasilacza. Czas ładowania wynosi około 2 godzin. Komputer może pracować bez doładowania baterii

RI

od 5 do 10 godzin. Konstruktorzy zastosowali rozwiązania, które służą oszczędności energii i mają umożliwić długotrwałą pracę bez konieczności ładowania baterii. Nowoczesne układy scalone zużywają bardzo niewiele energii. Wyczerpywanie się baterii jest odpowiednio wcześniej sygnalizowane przez system.

Komputer samoczynnie wygasza monitor, gdy użytkownik przez pewien czas nie wprowadza żadnych danych. STBook wyłącza się samoczynnie po zamknięciu pokrywy. Nie trzeba przy tym pamiętać o konieczności zapamiętania aktualnie opracowywanych informacji. Komputer sam troszczy się o dokładne zapisanie stanu w jakim w danej chwili się znajdował. Po ponownym otwarciu obudowy stan ten jest automatycznie odtwarzany. Użytkownik zastaje to samo środowisko oraz ten sam program i może natychmiast powrócić do przerwanej pracy.

STBook będzie sprzedawany w sklepach na Zachodzie prawdopodobnie jeszcze w tym roku. Cena wersji z 1 MB pamięci wyniesie około 3.000 DM

STPad

Kolejnym prototypem wystawianym przez Atari było urządzenie o nazwie STPad, które również jest interesującą odmianą elektronicznego notesu. STPad nie ma klawiatury ani myszy. Do wprowadzania danych służy ciekłokrystaliczny ekran i specjalne pióro, którym wskazujemy odpowiednie opcje menu. Rozdzielczość obrazu wynosi 640 x 400 punktów. Użytkownik może pisac na tym ekranie tak jak na zwykłym arkuszu papieru. Urządzenie jest w stanie rozpoznawać nawet pismo odręczne.

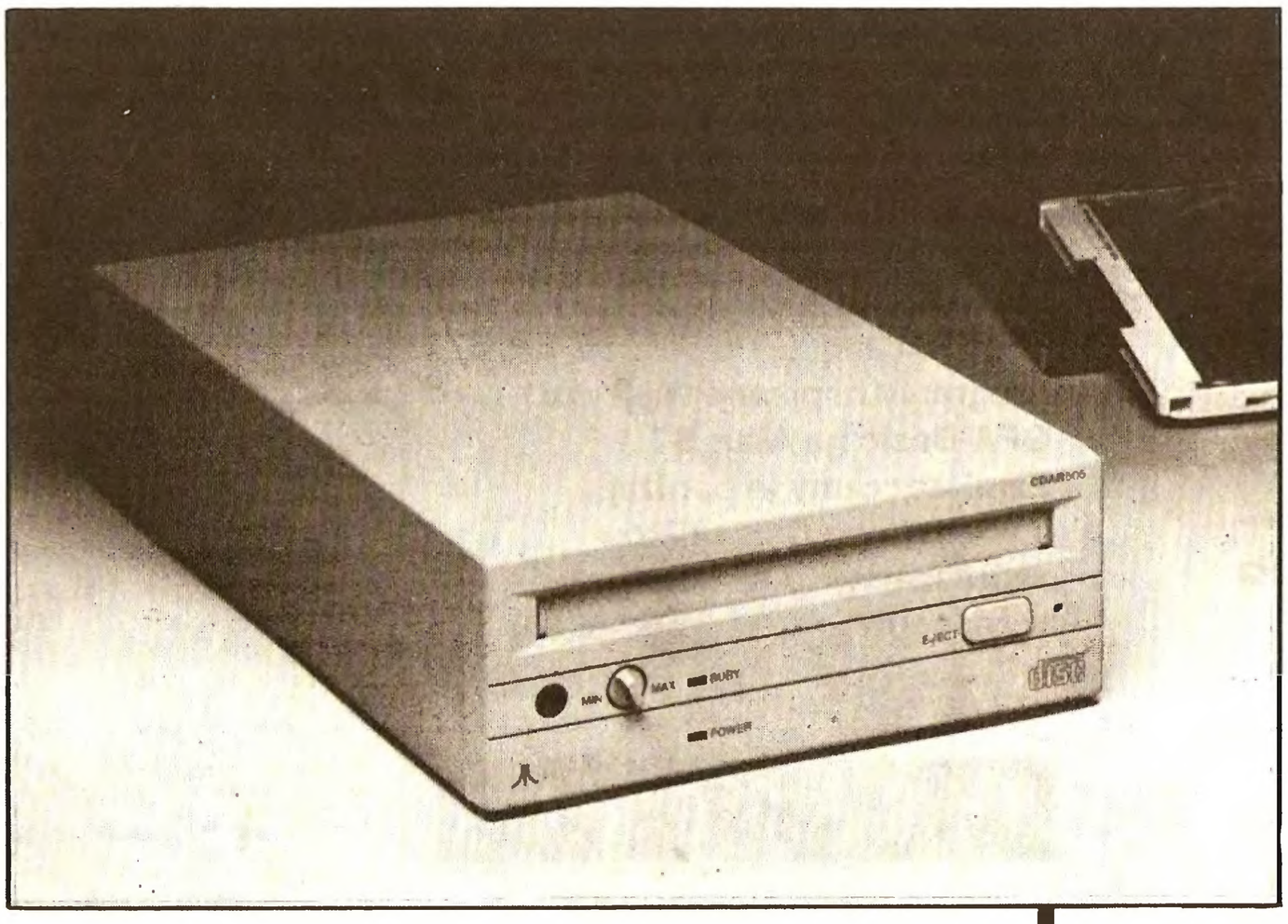
Baterie wystarczają na 10 godzin pracy. Konstrukcja STPad gwarantuje oszczędne zużycie energii. Po zakończeniu pracy komputer jest przełączany w stan spoczynku. Tu także nie jest potrzebne pamiętanie o konieczności zapisania opracowywanych danych. Troszczy się o to sam komputer. Po ponownym włączeniu odtwarzany jest obraz, który był widoczny przed wyłączeniem.

Do zapisu danych służy ciekawy i bardzo modny ostatnio rodzaj pamięci masowej. Informacje i programy, z których korzysta użytkownik, są przechowywane na specjalnych kartach ROM, lub zasilanych bateryjką kartach RAM. Pojemność takiego nośnika wynosi do 4 MB. Sposób zarządzania danymi zapisywanymi na kartach i ich struktura odpowiada strukturze zapisu na zwykłej dyskietce elastycznej lub twardym dysku. Komputer ma dwa gniazda do przyłączania kart.

CDAR505

Atari ma również inne znaczące osiągnięcia w dziedzinie rozwoju pamięci masowych wykorzystujących dyski kompaktowe. Na targach CeBIT wystawiono nowy rodzaj napędu dysków optycznych, który został oznaczony symbolem CDAR 505.

Urządzenie to można przyłączyć do komputera z rodziny STE lub TT. Ma ono pojemność 500 MB. Odczyt i zarządzanie danymi zapisanymi na dysku CD-ROM jest możliwe dzięki pakietowi oprogramowania o nazwie COBRA. Jeszcze jedna informacja, która ucieszy miłośników dobrej muzyki. Stacja dysków CD-ROM firmy Atari może być również wykorzystywana jako zwykły odtwarzacz płyt kompaktowych. Jego cena ma się kształtować poniżej 1.000 DM.



CDAR 505 to odtwarzacz płyt kompaktowych lub napęd dysków optycznych o pojemność 500 MB.

UNIX na Atari TT/030

Dla użytkowników 32-bitowego modelu Atari TT/030 posługujących się systemem UNIX opracowano pakiet procedur użytkowych w języku C i C++. Bardzo ciekawie wygląda środowisko graficzne bazujące na systemie UNIX System V Release 4.0 i X Window. Efekty pracy w tym środowisku widać na zdjęciu. Do demonstracji posłużył komputer z 19" monitorem TTM 194 o rozdzielczości obrazu 1280 x 960 punktów. Na uwagę zasługują także rozszerzenia pamięci, które pozwalają rozbudować Atari TT do 26 MB.

Janusz JARMOCH

DANE TECHNICZNE

STBook

* Procesor Motorola 68000
* Częstotliwość procesora 8 MHz
* Pamięć robocza 1 lub 4 MB
* Twardy dysk 20 MB
* Interfejs RS232, równoległy, 2xMIDI, DMA i FDD, gniazdo dodatkowej klawiatury numerycznej, złącze krawędziowe
* Klawiatura 84/85 klawiszy kompatybilna z STE/TT
* Wbudowany manipulator joypad
* Waga ok. 1 kg
* Czas pracy przy zasilaniu z baterii do 10 godzin
* Ekran ciekłokrystaliczny LCD 640 x 400 punktów
* Dodatkowe wyposażenia:
  - Modem telefaxu
  - Twardy dysk 40 lub 60 MB
  - Mysz
  - Zewnętrzna stacja dysków 1,44 BM
  - Klawiatura numeryczna

STPad

* Procesor Motorola 68000
* Częstotliwość procesora 8 MHz
* Pamięć robocza 1 lub 4 MB
* Karty RAM lub ROM o pojemności do 4 MB
* Interfejs RS232, równoległy, MIDI, DMA, złącze krawędziowe
* Waga 1,5 kg
* Czas pracy przy zasilaniu z baterii do 10 godzin
* Wymiary arkusza A4, grubość 36 mm
* Ekran ciekłokrystaliczny LCD 640 x 400 punktów
* Wprowadzanie danych poprzez ekran przy pomocy pióra
* Możliwość przyłączenia dodatkowej klawiatury typu Mega ST

UNIX i stacja robocza Atari TT/030 z monitorem TTM 194.

ST PROGRAMY

Wielu Czytelników zwraca nam uwagę, że zbyt mało miejsca poświęcamy komputerom ST. W pełni zgadzamy się z tym zarzutem, a usprawiedliwić może nas w pewnym stopniu tylko brak autorów. Na szczęście otrzymaliśmy już pierwsze, proste programy napisane w języku GFA-Basic na Atari ST. Zamieszczamy je poniżej, a wszystkich Czytelników zapraszamy do współpracy w redagowaniu naszego pisma.

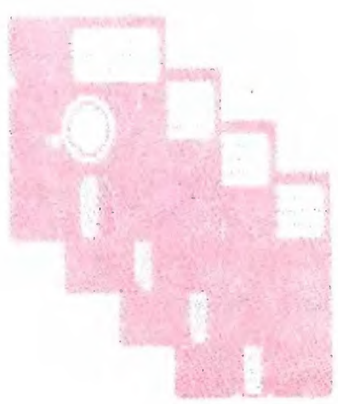
(red.)

HISTOGRAM

Program ten służy do sporządzania na ekranie wykresu słupkowego 20 danych. Po uruchomieniu podajemy dane, a program rysuje wykres i obok niego umieszcza wprowadzone wartości liczbowe.

Tomasz Gawęda

```
' HISTOGRAM
' Tomasz Gaweda
' (c) 1991, Sp. Bajtek
'
DIM x(20)
DEFLINE 3
po:
DEFTEXT 1,,,4
CLS
a1=0
FOR i=1 TO 20
  PRINT "dana nr.";i;" ";
  INPUT a
  x(i)=a
  IF a>a1
    a1=a
  ENDIF
NEXT i
IF a1>0 AND a1<19
  d=10
ENDIF
IF a1>19 AND a1<190
  d=1
ENDIF
IF a1>190 AND a1<1900
  d=0.1
ENDIF
IF a1>1900 AND a1<19000
  d=0.01
ENDIF
IF a1>19000
  GOTO po
ENDIF
ry:
CLS
DRAW 0,190 TO 300,190
FOR z=190 TO 0 STEP -10
  DRAW 0,190-z TO 300,190-z
  TEXT 300,z,190/d-z/d
NEXT z
FOR i=1 TO 20
  PBOX i*10+5,190,1*10+10,190-x(i)*d
NEXT i
DEFTEXT 1,,,6
FOR i=1 TO 20
  LOCATE 50,i
  IF i<10
    PRINT 0;
  ENDIF
  PRINT i;" ___ ";x(i)
NEXT i
DEFTEXT 1,,,7
TEXT 400,180,"Jeszcze raz ? (t/n)"
s=INP(2)
PRINT s
IF s=116
  FOR i=1 TO 20
    x(i)=0
  NEXT i
  GOTO po
ELSE
  CLS
  END
ENDIF
```



RÓWNANIE KWADRATOWE

Szybkie obliczenie pierwiastków rzeczywistych równania kwadratowego umożliwia następny program. Wystarczy tylko podać współczynniki a, b i c równania postaci $ax^2+bx-c=0$, a natychmiast otrzymujemy wynik.

Tomasz Gawęda

```
' ROWNANIE KWADRATOWE
' Tomasz Gaweda
' (c) 1991, Sp. Bajtek
'
PRINT "Dane jest rownanie Ax^2+Bx+C=0"
PRINT
INPUT "Podaj A ";a
INPUT "       B ";b
INPUT "       C ";c
IF a=0
  RUN
ENDIF
delta=b*b-4*a*c
IF delta<0
  PRINT
  PRINT "Rownanie nie ma rozwiazania."
  END
ELSE
  IF delta>0
    x1=(-b-SQR(delta))/(2*a)
    x2=(-b+SQR(delta))/(2*a)
    PRINT
    PRINT "Rozwiazaniami rownania ";
    GOSUB ro
    PRINT " sa :"
    PRINT "x1 = ";x1
    PRINT "i"
    PRINT "x2 = ";x2
  ELSE
    x0=-b/2*a
    PRINT
    PRINT "Rozwiazaniem rownania ";
    GOSUB ro
    PRINT " jest :"
    PRINT "x0 = ";x0
  ENDIF
ENDIF
END
PROCEDURE ro
  PRINT a;"x^2+";b;"x+";c;"=0";
RETURN
```

WYKRES FUNKCJI

Kolejny program służy do rysowania przebiegu dowolnej funkcji. Wzór tej funkcji należy umieścić w programie przed jego uruchomieniem. Program może być szczególnie przydatny podczas odrabiania zadań z trygonometrii. Zakończenie programu i przejście do interpretera następuje po naciśnięciu dowolnego klawisza.

Konrad Giżycki

Marcin Giżycki

```
' WYKRESY FUNKCJI
' Konrad Gizycki
' (c) 1991, Sp. Bajtek
'
CLS
DEFTEXT 1,,,6
COLOR 1
@ukl
FOR x=-2*PI TO 2*PI STEP 0.02
  ' ****************************
  '
  y=2*SIN(3*x)-4*COS(x/2) ! tu wpisz wzor funkcji
  '
  ' ****************************
  x1=x*20+300
  y=100-y*20
  COLOR 1
  PLOT x1,y
  EXIT IF INKEY$<>""
NEXT x
DO
  EXIT IF INKEY$<>""
LOOP
END
PROCEDURE ukl
  TEXT 250,195,"by MG & KG"
  TEXT 620,110,"x"
  TEXT 275,15,"y"
  LINE 300,10,300,300
  LINE 300,10,295,15
  LINE 300,10,305,15
  LINE 10,100,620,100
  LINE 615,98,620,100
  LINE 615,103,620,100
  FOR a=20 TO 180 STEP 20
    LINE 295,a,305,a
  NEXT a
  FOR a=0 TO PI*40+1 STEP PI/10
    LINE 300+a,95,300+a,105
    LINE 300-a,95,300-a,105
  NEXT a
  DEFTEXT 1,,,7
  TEXT 300+PI*40,110,"2"+CHR$(227)
  DEFTEXT 1,,,6
  TEXT 290,83,"1"
RETURN
```

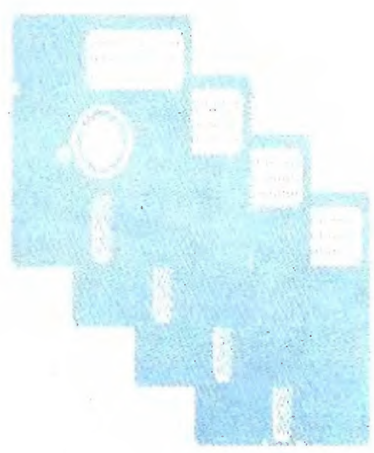
DOKŁADNE DZIELENIE

Niekiedy potrzebny jest bardzo dokładny wynik dzielenia, a dokładność obliczeń wykonywanych przez komputer jest ograniczona. W takim wypadku można użyć programu, który zapewnia nieograniczoną dokładność wykonywania tej operacji. Po uruchomieniu tego programu wpisujemy dzielną i dzielnik. Komputer oblicza i wyświetla wynik, a w lewym górnym rogu podawana jest aktualna liczba miejsc po punkcie dziesiętnym, czyli dokładność obliczenia. Zatrzymanie i wznowienie programu uzyskujemy przez naciśnięcie klawisza spacji, jego ponowne uruchomienie zaś przez naciśnięcie ‹P›. Klawisz ‹Esc› powoduje przerwanie programu i powrót do interpretera.

Konrad Giżycki

Marcin Giżycki

```
' DZIELENIE
' Konrad Gizycki
' (c) 1991, Sp. Bajtek
'
DEFFILL 0
@prg
END
PROCEDURE prg
  i1=0.1
  INPUT a
  INPUT b
  c=INT(a/b)
  PRINT c;
  IF c=a/b
    PRINT
    @prg
  ENDIF
  PRINT ".";
  a=(a-c*b)*10
  REPEAT
    i=i+1
    l=LEN(STR$(i))
    IF i/80=INT(i/80) OR l>l1
      l1=l
      PBOX 1,1,l*8+5,12
      BOX 1,1,l*8+5,12
    ENDIF
    TEXT 4,9,1
    c=INT(a/b)
    PRINT c;
    a=(a-c*b)*10
    @key
    IF k=27
      END
    ELSE
      IF k=80 OR k=112
        RUN
      ELSE
        IF k=32
          @czek1
        ENDIF
      ENDIF
    ENDIF
  UNTIL a=0
  @czek2
RETURN
PROCEDURE key
  IF BIOS(1,2)
    k=INP(2)
  ELSE
    k=-1
  ENDIF
RETURN
PROCEDURE czek1
  REPEAT
    @key
  UNTIL k<>32
  REPEAT
    @key
  UNTIL k=32
RETURN
PROCEDURE czek2
  REPEAT
  UNTIL INKEY$<>""
  RUN
RETURN
```



Od paru lat komputery Macintosh zaliczane są do urządzeń najwyższej klasy. Niestety, ich cena znacznie przekracza możliwości finansowe wielu użytkowników. Nic dziwnego, że w takiej sytuacji wiele firm software'owych podjęło prace nad opracowaniem

EMULATOR MACINTOSHA

znacznie tańszych emulatorów. Macintosh natychmiast wypowiedział wojnę konkurencji zagrażającej jego interesom. Parę spraw skierowano do sądu. Producenci Aladina, jednego z pierwszych emulatorów Apple działającego na Atari ST, zostali zmuszeni do wycofania swojego produktu z rynku.

Na szczęście nie wszystkie sprawy miały tak przykre zakończenie. Spory prawne wokół innego emulatora Macintosha, który nazywa się **Spectre 128,** zostały pomyślnie rozstrzygnięte dla jego twórców. Dla użytkowników ST jest on bardzo atrakcyjnym rozwiązaniem. Obecnie oferowana na Zachodzie wersja 3.0 kosztuje zaledwie ok. 570 DM. Przy tak niskiej cenie emulatora trudno się dziwić irytacji i niezadowoleniu producentów komputerów Apple. Każdy oryginalny egzemplarz Macintosha kosztuje przecież kilkakrotnie więcej.

Konstruktorem emulatora **Spectre** jest Dave Small z USA. Prace nad tym urządzeniem trwają już od dłuższego czasu. Poprzednie wersje nosiły nazwę **Spectre 128** i **Spectre GCR.** Użytkownicy pierwszej z nich nie mogli bezpośrednio czytać dyskietek Macintosha. Jedynym sposobem przeniesienia danych z Apple na ST było wówczas połączenie obu komputerów kablem i przegranie przy pomocy specjalnego oprogramowania użytkowego. **Spectre GCR** umożliwił już czytanie dyskietek. Trzyliterowy skrót w nazwie pochodzi od formatu zapisu danych na dyskietkach Macintosha (Group Code Recording). Aby przetłumaczyć taki zapis na format zrozumiały dla ST potrzebny specjalny układ logiczny kontrolera stacji dysków.

Spectre 128 wersja 3.0 zawiera kolejne udoskonalenia. Składa się ona z modułu ROM przyłączanego do portu ROM komputera, kabla łączącego moduł z wewnętrzną lub zewnętrzną stacją dysków elastycznych Atari ST, dwóch dyskietek z drajwerami emulatora i programami Public-Domain oraz 160-stronicowej instrukcji obsługi. Do uruchomienia **Spectre 128** potrzebny jest jeszcze zamawiany oddzielnie zestaw kości ROM o łącznej pojemności 128 KB, który zawiera system operacyjny Macintosha, oraz dyskietki systemowe. ROM-y z systemem operacyjnym wkłada się do specjalnych gniazd w module ROM. Potem cały moduł należy połączyć kablem ze stacją dysków. W module ROM znajduje się układ logiczny kontrolera dysków elastycznych, który pozwala bezpośrednio odczytywać i zapisywać dyskietki w formacie Macintosha na stacji dysków ST.

Emulator akceptuje wszystkie wersje systemu operacyjnego Macintosh do numeru 6.0.5. Niektórzy twierdzą też, że wersja 7.0 również nie powinna przysparzać kłopotów.

Procedury **Spectre 128** zajmują 256 KB w pamięci operacyjnej Atari ST. Oprócz tego potrzebny jest także pewien spory obszar na sam system operacyjny Macintosha. W ST z 1 MB pamięci pozostaje tak niewiele miejsca, że możliwe jest korzystanie z programów Macintosha w bardzo ograniczonym zakresie. Wskazane jest więc posługiwanie się komputerem z co najmniej 2 MB RAM. Zadowalające efekty podczas pracy uzyskuje się, jeżeli ST jest wyposażone w 2 stacje dysków elastycznych, a najlepiej w 1 stację dysków elastycznych i twardy dysk.

Możliwe jest także sformatowanie jednej lub kilku partycji twardego dysku Atari jako partycji Mac'a. **Spectre 128** współpracuje także z napędami dysków wymiennych, mp. Megafile 44 MB lub kompatybilnymi. Stanowi to szczególnie dogodny sposób wymiany danych pomiędzy Apple a Atari ST.

Rozdzielczość podstawowych trybów graficznych ST jest wyższa niż w Macintoshu. Nie stwarza to problemów, ponieważ oprogramowanie jest zwykle pisane w sposób uniezależniający je od rozdzielczości. Niestety użytkownicy **Spectre 128** muszą zrezygnować z trybu kolorowego oraz z wykorzystywania układów OverScan lub HyperScan, które podwyższają rozdzielczość obrazu.

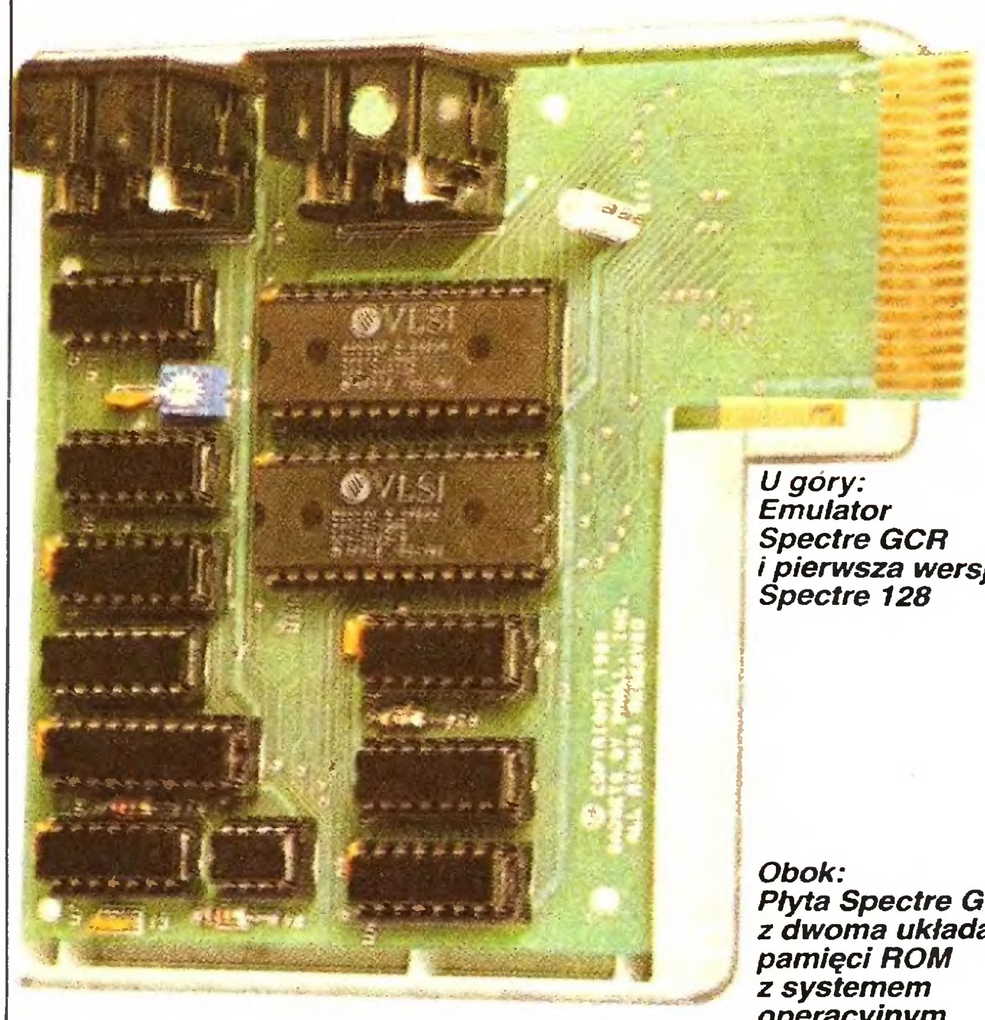
Emulator znakomicie naśladuje dźwięk Mac'a. Pewne spowolnienie i nieznacznie gorsze brzmienie słyszy się tylko w przypadku programów bardzo intensywnie korzystających z możliwości dźwiękowych komputera, np. programu Jamsession.

Począwszy od wersji 3.0 można instalować **Spectre 128** także w komputerach Atari TT. Nie emuluje on jednak 32-bitowych Macintoshy z procesorem 68030. Koprocesor matematyczny 68882 jest wprawdzie rozpoznawany, ale nie jest w pełni wykorzystywany przez oprogramowanie.

W kolejnych wersjach emulatora **Spectre 128** zapowiadane jest opracowanie rozwiązań umożliwiających pełne wykorzystanie drukarki laserowej Atari-Laser, dopasowanie do kart graficznych ST o wysokiej rozdzielczości, lepsza współpraca z rozszerzeniami sprzętowymi ST, a także skonstruowanie złącza szeregowego standardu SCSI do urządzeń peryferyjnych Macintosha.

Janusz JARMOCH

Literatura:
1) Christoph Dernbach, „Ganz nah am Original", ST Magazin 1/90
2) „Die ST Apfelkur", ST Magazin 5/91
3) Roger Howorth, „Changes for the better", Personal Computer World 11/89



U góry: Emulator Spectre GCR i pierwsza wersja Spectre 128

Obok: Płyta Spectre GCR z dwoma układami pamięci ROM z systemem operacyjnym Macintosha.

dokończenie ze str. 32

Obudowa komputera niczym nie przypomina tradycyjnych ST. Wygląda ona raczej jak jakieś skomplikowane, elektroniczne urządzenie pomiarowe, z którym możemy zetknąć się w laboratoriach naukowych. Poważny wygląd nadają mu widoczne z przodu, wysuwne, 19" panele. Dwa uchwyty na bocznych ściankach obudowy ułatwiają transport dość ciężkiego komputera.

100 MB NA KARCIE

Dodatkowe karty rozszerzające są również dostosowane do standardu przemysłowego. Pamięci masowe stosowane przez IBP mają postać wysuwnych, łatwych w montażu i demontażu paneli. Skonstruowanie stacji dysków elastycznych na karcie nie nastręczało większych problemów. Przy twardym dysku występują problemy termiczne. Obok dysku i interfejsu SCSI umieszczono więc dodatkowy, miniaturowy wentylator, który chroni całe urządzenie przed nadmiernym wzrostem temperatury. W zależności od potrzeb użytkownik może kupić do swojego komputera twardy dysk 3,5" o pojemności od 32 do 100 MB.

Dla profesjonalisty bardzo ważnym kryterium oceny walorów systemu jest także możliwość zabezpieczenie danych. Konstruktorzy przemysłowego ST pomyśleli także o tym i przewidzieli odpowiedni sposób ochrony informacji przetwarzanych przez użytkownika. Jednym z elementów komputera 190 ST 020 może być streamer, który jest przyłączany przez złącze DMA lub SCSI. Służy do sporządzania zapasowych kopii zbiorów.

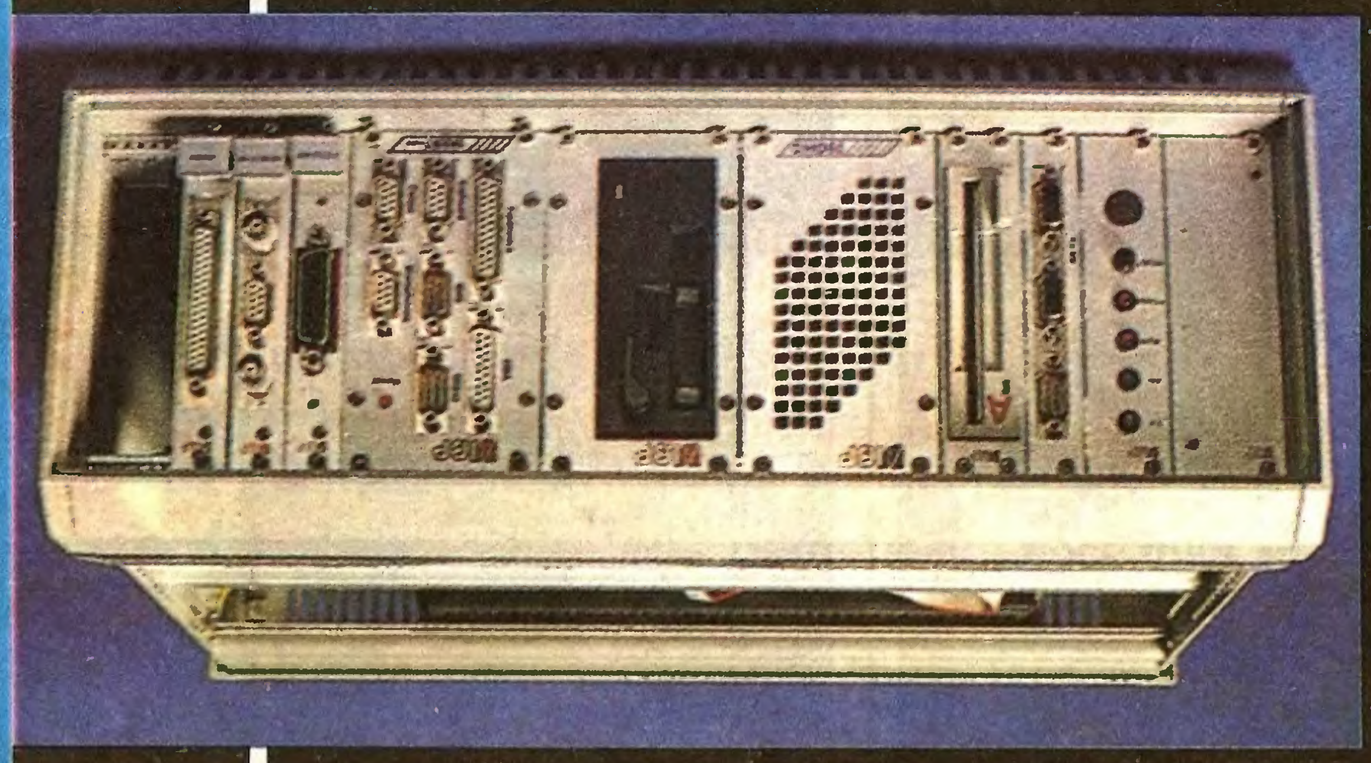
KONFIGURACJA NA ŻYCZENIE

Wraz z komputerem dostarczane są dyskietki z oprogramowaniem systemowym. Znajduje się tam między innymi program, który potrafi zidentyfikować konfigurację kart zastosowanych w całym systemie.

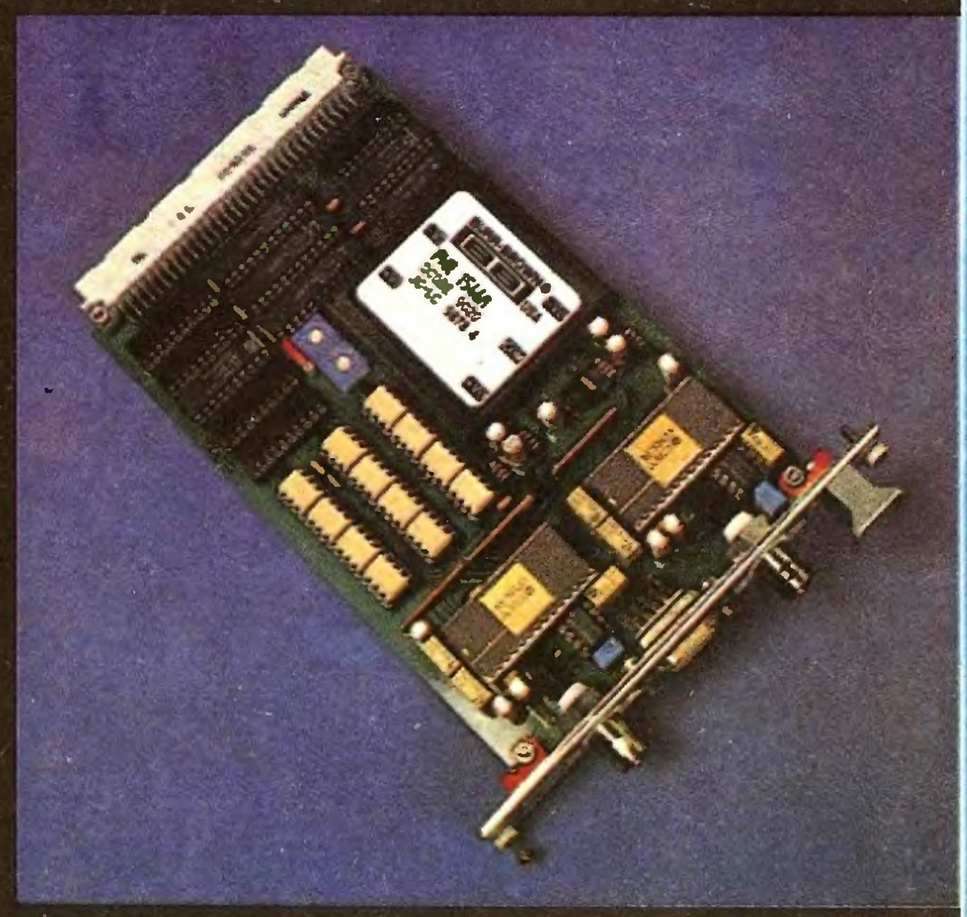
Dokumentacja techniczna komputera firmy IBP jest bardzo bogata. W solidnym segregatorze zebrano wyczerpujący, dwujęzyczny, angielsko-niemiecki opis systemu i całego hardware serii 190 ST. Użytkownik znajdzie tu bardzo dokładne wskazówki dotyczące rozbudowy komputera (rozszerzenia pamięci, instalacji koprocesora, itd).

Konfigurację komputera serii 190 ST można bardzo łatwo dopasować do wymagań i potrzeb użytkownika. Konstrukcja systemu jest bardzo uniwersalna i elastyczna. Komputer znakomicie nadaje się do wykorzystania jako urządzenie sterujące i regulacyjne we wszystkich gałęziach przemysłu. Jego cena zależy od konfiguracji. Najtańszą wersję można kupić już za 5330 DM.

Janusz JARMOCH
opracowanie na podst. ST-Magazin 6/91



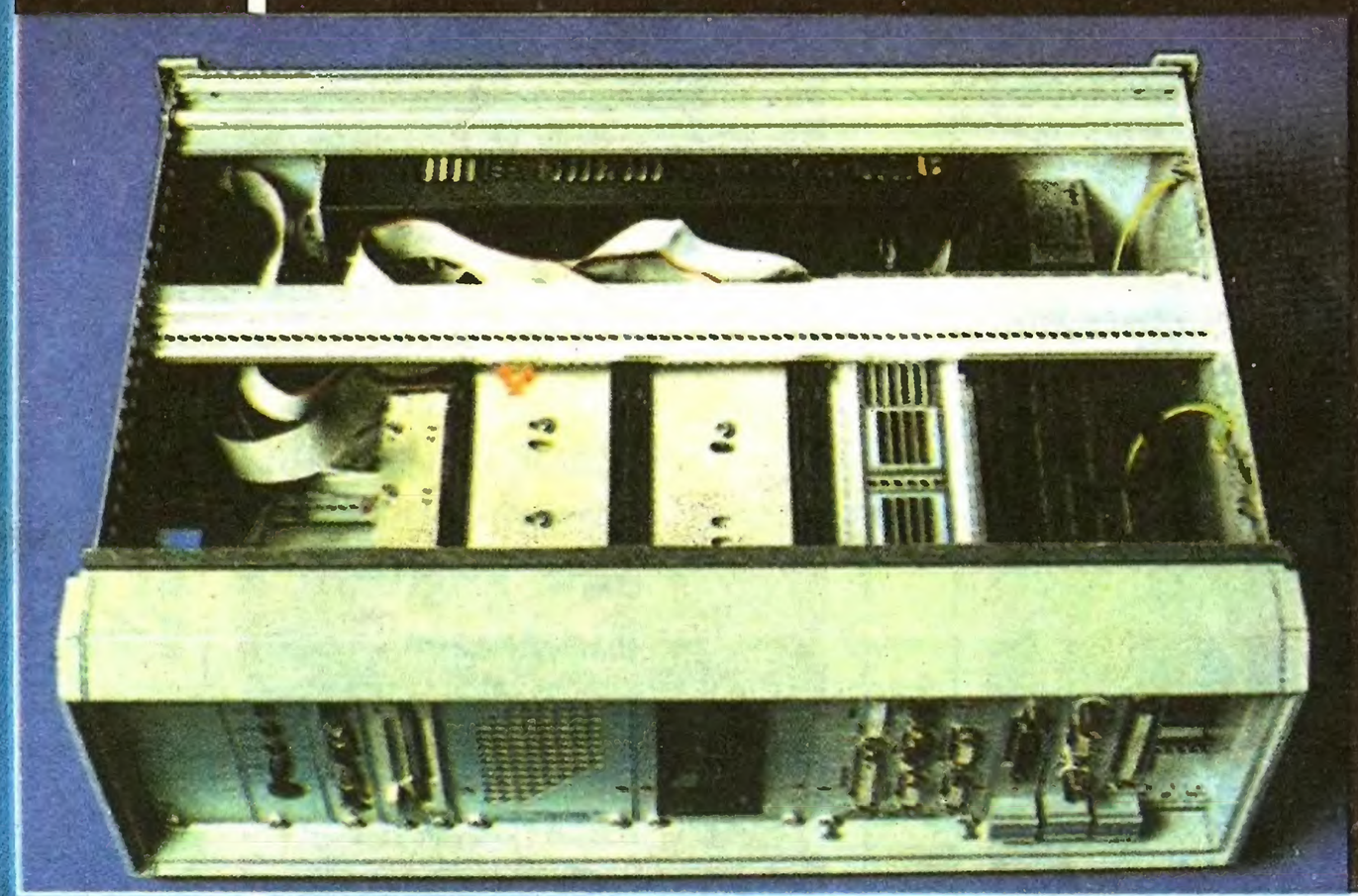
Panele przemysłowego komputera 190 ST 020.



Jedna z wysuwanych kart komputera 190 ST 020.

ST DLA PROFESJONALISTY

ST może być wykorzystywane w bardzo wielu dziedzinach. Z Atari korzystają naukowcy, muzycy, graficy, dziennikarze. Ale standardowa wersja tego komputera nie spełnia wymagań stawianych systemom przemysłowym. Komputery przeznaczone do pracy w warsztatach produkcyjnych są konstruowane według zupełnie innych reguł. Ich wygląd w niczym nie przypomina urządzeń, które stoją w domach i biurach.



Wnętrze komputera.

STANDARD PRZEMYSŁOWY

Tylko nieliczni sympatycy Atari w Polsce wiedzą, że istnieje także ST w wersji przemysłowej. Firma IBP oferuje taki komputer wykonany w postaci 19" kaset zgodnych z europejskimi standardami przemysłowymi. Najnowsza wersja przemysłowego ST została oznaczona symbolem 190 ST 020.

Pierwsze komputery ST nie nadawały się wykorzystania jako element profesjonalnych systemów przemysłowych. Przyczyną tego były niedogodne wymiary płyty głównej i brak możliwości przyłączenia urządzeń peryferyjnych niezbędnych w profesjonalnym laboratorium. Firma IBP postanowiła wypełnić tę lukę i skonstruowała własny, zupełnie nowy komputer zgodny z ST. Składa się on z 3 kart o wymiarach odpowiadających europejskim standardom. Wszystkie złącza komputera znajdują się na przedniej ściance wysuwanej kasety. W warunkach przemysłowych stwarza to znacznie dogodniejsze warunki do przyłączania urządzeń zewnętrznych. Niektóre ze złącz nieco odbiegają od standardów ST, za to odpowiadają normom powszechnie stosowanym w aparaturze kontrolno-pomiarowej i sterującej. Gwarantuje to także użytkownikowi możliwość łatwego przyłączenia standardowych monitorów i klawiatur.

WYPOSAŻENIE

Konstruktorzy przemysłowego ST opracowali również szereg ciekawych, dodatkowych elementów wyposażenia komputera. Wśród opcji oferowanych użytkownikowi znajduje się rozszerzenie z 512 KB do 2 MB, wzmacniacz audio o mocy 1 W, zegar czasu rzeczywistego zasilany z baterii, złącze DMA, interfejs MIDI o szybkości przesyłania danych 126 kBodów, złącze V.24 do przyłączenia klawiatury, gniazda do przyłączenia kart RAM lub ROM z oprogramowaniem.

MOTOROLA 68020

Komputery serii 190 są wykonywane w trzech zasadniczych wersjach. Pierwsza z nich to normalny, kompatybilny z ST komputer, który posiada procesor 68000, taktowany z częstotliwością 8 MHz, blitter i gniazdo do przyłączenia koprocesora arytmetycznego. Druga z wersji jest oznaczona symbolem 190 ST 30 i zawiera dodatkowo procesor V30, który otwiera drogę do wykorzystywania oprogramowania MS-DOS. Trzecia odmiana jest oferowana pod nazwą 190 ST 020. Najważniejszym elementem komputera w tej wersji jest trzydziestodwubitowy, traktowany z częstotliwością 16 MHz procesor Motorola 68020. Pozwala to osiągnąć dużą szybkość działania programów, w których wykonywane są złożone obliczenia.

Komputery 190 ST 020 pracują pod kontrolą specjalnego systemu operacyjnego, który gwarantuje odpowiednie dopasowanie do większości typowych programów. Można na nim uruchomić różne gry, programy graficzne, program CAD „Dyna-CAD", program do projektowania płytek układów elektornicznych „Platon". W tym przypadku szczególnie uwidaczniają się zalety szybkiego procesora 68020. Różnica szybkości działania pomiędzy 190 ST 020 a zwykłym ST jest porównywalna do różnicy pomiędzy komputerom klasy 386 i zwykłym XT.

dokończenie na str. 31